

国际信息工程先进技术译丛



基于4G系统的移动服务技术

Towards 4G Technologies: Services with Initiative

（德） Hendrik Berndt　主编

彭　晋　段晓东　张云飞
吴亦川　廖洪銮　张剑寅　译
佘春燕　李　洋　周乃宝





机械工业出版社
CHINA MACHINE PRESS

VISIT…

LANZAROTE
Caliente.com

关于4G

4G技术为下一代移动通信系统中正在涌现的未来网络和服务提供技术提供了一个全面的解释。创造者们解释了个性化服务、移动P2P、语义计算和智能上下文等新技术对现有技术的取代，同时它们也是实现未来创新服务的必要条件。本书介绍了下一代移动服务提供中的最新挑战和机遇，并定义了一些超越个人移动性和移动个性化的新模式，如基于用户偏好的服务发现，灵活、位置感知自适应的服务。

这宝贵的指南将帮助R&D部门的电信工程师、首席执行官、电信管理员以及电气电子工程和通信领域的学术研究人员更全面地了解下一代移动系统的技术和服务。

迈向4G技术：

- 全面地描述未来网络和服务的提供技术。
- 涵盖热点话题讨论，如智能的用户配置、主动服务选择、基于上下文感知的服务提供和泛在计算。
- 在开放的可编程体系架构中引入多样化技术。
- 显示它们如何协作，以给用户全新的体验。
- 提供案例学习，以便更好地理解理论。



国际信息工程先进技术译丛

基于4G系统的移动服务技术

（德）Hendrik Berndt　主编
彭　晋　段晓东　张云飞
吴亦川　廖洪銮　张剑寅　译
佘春燕　李　洋　周乃宝



机 械 工 业 出 版 社

本书介绍了构建一种前所未有的新的业务提供的方法学、解决方案以及富有前景的深入视角。通过对未来网络和业务特征的介绍，以及对4G移动通信系统中的关键技术的探讨，开发出4G移动环境感知业务，使人们向一个新的业务空间迈进。

本书适合于从事通信产品开发和网络规划设计的广大工程技术人员，也可作为高等院校通信、计算机等专业在校师生的参考书籍，对于P2P、中间件等技术的研究者也同样适用。

Towards 4G Technologies：Services with Initiative/by Hendrik Berndt
ISBN：978-0-470-01031-0
All Rights Reserved. This translation published under license.
Original English language edition copyright © 2008 by John Wiley & Sons Ltd.
Simplified Chinese Translation Copyright © 2010 by China Machine Press.
本书中文简体翻译出版授权机械工业出版社独家出版，并限定在中国大陆地区销售，未经出版者书面许可，不得以任何方式复制或发行本书的任何部分。
本书封面贴有Wiley公司的防伪标签，无标签者不得销售。
本书版权登记号：图字01-2008-3264号

图书在版编目（CIP）数据

基于4G系统的移动服务技术/(德)伯尼特（Berndt，H.）主编；彭晋等译.—北京：机械工业出版社，2010.1
(国际信息工程先进技术译丛)
Towards 4G Technologies：Services with Initiative
ISBN 978-7-111-29117-6

Ⅰ.基…　Ⅱ.①伯…②彭…　Ⅲ.移动通信－通信技术　Ⅳ.TN929.5

中国版本图书馆CIP数据核字（2009）第217221号

机械工业出版社（北京市百万庄大街22号　邮政编码100037）
策划编辑：张俊红　责任编辑：朱　林　版式设计：霍永明
封面设计：马精明　责任校对：闫玥红　责任印制：杨　曦
保定市中画美凯印刷有限公司印刷
2010年2月第1版第1次印刷
169mm×239mm·17.5印张·338千字
0001—3000册
标准书号：ISBN 978-7-111-29117-6
定价：78.00元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

电话服务
社服务中心：（010）88361066
销 售 一 部：（010）68326294
销 售 二 部：（010）88379649
读者服务部：（010）68993821

网络服务
门户网：http：//www.cmpbook.com
教材网：http：//www.cmpedu.com
封面无防伪标均为盗版

译 者 的 话

第3代移动通信的高带宽特性给人们带来了全新的多媒体服务感受，3GPP R5版本定义的IP多媒体子系统（IMS）使用SIP来控制、管理多媒体服务，为用户提供了良好的使用体验，但它仍然没有突破传统服务提供方式的局限。未来的移动服务以泛在环境为基础，它们的无线接入网速度各异，用户终端处理能力参差不齐，而用户对服务的要求却普遍提高，这一切都使得3G技术已不能满足当前的需求。

4G移动通信系统为移动通信服务带来了更为先进的理念和技术，使得应用和用户成为移动通信的主角。它应用了正在涌现的个性化服务、移动P2P、语义计算、上下文感知等全新技术，使服务可以根据环境的变化从无线传输速率到应用进行调整，因此具有更强的扩展能力和适应能力。

由于国内各大组织及运营商的积极推进，3G网络正迅速走进人们的生活，4G技术也已随着科技前进的步伐进入到人们的视野。本书对4G移动通信系统中的服务特性进行了详细描述，指出未来的服务将是以用户为中心的个性化服务；同时也对目前新兴的技术和服务平台进行了客观的分析和研究，以全新的视角提出了一种前所未有的新的服务提供方法学。

本书内容全面，不仅对未来的网络和服务特征进行详细阐述，对4G移动通信系统中的关键技术也进行了深入的探讨。本书适合于从事通信产品开发和网络规划设计的广大工程技术人员阅读，也可作为高等院校通信、计算机等相关专业师生的参考书，对于P2P、中间件等技术的研究人员也同样适用。

书中涉及大量的专业术语，翻译时尽量采用了国内权威出版专著中的译名或行业通用译名。为方便阅读，大多数术语均随译名在正文中一并列出。在本书最后部分给出了相关的延伸读物，感兴趣的读者可以进一步阅读研究。

本书全部由中国移动通信有限公司研究院多年从事移动通信领域研究的专家和工程技术人员翻译。其中彭晋负责前言和第1、2章的翻译；段晓东负责第7章的翻译；佘春燕负责第3章的翻译；张云飞负责第4、5章的翻译；吴亦川负责第6章的翻译；张剑寅负责第8、9章的翻译；廖洪銮负责第10、11章的翻译；李洋负责第12、13章的翻译。全书由佘春燕负责统稿。本书在翻译过程中

得到了研究院许多同事的大力帮助，在此表示衷心感谢！

限于译者水平，以及目前国内缺乏一些术语标准译名的现状，书中不妥之处，恳请广大读者批评指正。

译者

2010年春于北京

序　一

著名的丹麦物理学家 Niels Bohr 曾经指出："预言是困难的，尤其是对未来的预言"。然而，我们这个时代的许多思想领袖都在努力地尝试改变这种情况，因为世界上所有有趣系统的潜在的非线性特性使得我们在不断地思考、探索和创造未来，这些努力都是希望能产生正确的预言的行为。电信和计算领域的预言显得尤为糟糕。我最喜欢的一则引言曾经于 1949 年发表在美国的《Popular Mechanics》杂志上："将来的计算机的重量将不会超过 1.5t"。的确没错，因为从今天看来，很难想象我们能找到一台超过 1.5t 的计算机！

然而，预言仍是非常重要的，尤其是在电信领域与计算领域慢慢融合的部分。大约 40 年前，网络这个名词代表与电信领域相关的一些事物，而现在它却被计算领域的专家经常使用。同样在 40 年前，新兴的数字化计算领域与电信网络及其服务还毫不相关，而在今天，所有的交换节点都是数字化的，所有的计算机都接入网络，所有不同质量的服务、计费模型的融合和工程学可靠模型都使得电信/计算领域都不得不重新思考服务是什么，谁是被服务者。

我们的工业有大步跨越到所谓的"4G"的趋势，而这正是我们所处的世界。你可能找不到比本书更好的资源来找出"4G"究竟意味着什么，以及为什么你必须了解这种电信/计算融合的体系结构，移动和随时连接的网络，语义和本体以及智能代理。有一天，20 世纪 50 年代的电信工程找到了通往计算的道路；有一天，20 世纪 70 年代的智能的、具有语义感知能力的系统也会找到它们通向网络的道路。这本书就是为你知识的更新以及意识到上述这些事情将在什么时候发生而准备的。

美国的棒球运动员 Yogi Berra 同样也理解预测未来的困难性，他说道："未来总是和过去的不同，并且永远会和过去不同"。我们到某个时候会对未来的认识更深，但是它往往与我们所期望的不尽相同，然而我们至少相信基础科技能够构筑未来，所以我们能够及时地发明和创造我们的未来。同时，这本书将使你们的思想关注于这些基础是如何来支撑创新的服务在 4G 乃至以后网络上的传递的，而这些会改善我们的生活。

学习本书的知识会使你在未来的工作中游刃有余。

Dr Richard Mark Soley
主席，首席执行官
对象管理集团

序　　二

首先祝贺这本有关服务提供方面的最新进展和深入剖析的书籍的出版。这本书向读者呈现了未来移动服务的机遇和挑战，指明了面向4G技术的发展方向，它是网络和服务提供的非平行化发展的产物。

作者从普适服务和网络环境的视角介绍了移动通信网络的最新概念是如何来支撑未来的移动服务环境的。这个新的环境是新的服务概念构建所赖以存在的开放性基础平台，包括基于偏好的服务发现以及选择的方法，深层次的个性化和移动点对点的应用，以及语义增强和上下文智能的服务。这些原理在非平行的面向条件感知和自适应服务的维度上都超过了现有的方法。这本书分为3个主要部分：一个新的体系结构、智能服务提供的基础和智能服务在环境中的嵌入。因此，这本书涵盖了成功服务传递的方方面面。

本书所描述的技术发展代表了很大一部分的技术革新，但是作者并不想使读者们误以为他们并不需要关心和考虑这些革新。只有当成功的商业案例构建出来，才允许考虑多种团体和个人在移动信息社会中找到他们各自的商业角色。服务提供仍然在大范围内适用，所以消费者可以在任何旅游地来使用他们熟悉的服务。主动服务指的就是根据用户的喜好和行为模式提供服务，这也被希望成为新的用户的首选，因为它指明了一种信息的易于使用、方便实用的方式。

本书详细讨论了许多潜在的技术，从大系统的优化，到表示服务的基于语义网络的本体设计，再到面向上下文的编程等。我相信，这本书的知识结构能够使电信工程师、研发人员、电信管理者以及计算机领域、电子和电气工程领域、电信领域的科研人员受益。它为读者提供了有关下一代移动服务网络的全面诠释。

希望大家喜欢这本书！

Dr Atsushi Murase
NTT DoCoMo 研究实验室管理总监

前　　言

出版一本致力于提供包含绝大多数相关技术的高级服务的书籍是非常必要也是非常具有挑战性的事情。说必要是因为只有新的并且是空前的服务产品才有能力创造新的利润，而这些是所有运营商所积极寻求用来缓解日益加剧的市场竞争压力的方法。说其具有挑战性，是因为如何看待未来的服务，以及为了满足用户需求应具备的能力方面，每位读者有自己独到的想法。本书的目标就是讲述未来服务传递的方方面面，以及为服务选择、触发和服务运行时的条件智能添加空间。它展现了一条通向新的服务领域的道路。贡献者们沿着这条道路进行他们的移动探险。新一代的移动服务目标是通过灵活的服务以及基于可用的资源来协助用户进行日常生活。新的服务需要适应用户环境的改变并且在用户使用服务的过程中要感受到引导以及保护的存在。许多相关部门需要完成4G技术的规划，在这当中存在一个底层的多元化互联和一个开放的可编程的体系架构，它们可以为系统进行重新配置和优化。这些相应的组件将在本书的后续章节中进行详细讲述。并且，它遵循下一代服务架构的高级服务提供平台以及由此而扩展的一个普遍适应的服务环境也将在本书中进行介绍。本书还将会介绍一个新的服务提供体系，它是主动的请求服务而不是相对麻烦的服务发现、非优化的服务选择，也不是有限的用户指引以及不方便的服务执行。

本书特色

本书第一次对下一代移动系统中的未来网络和服务传递所需要的技术进行了全面的综述。它对如何个性化、点对点的解决方案、语义计算、本体论工程以及适合新概念的逻辑系统描述进行了深入的介绍。同时本书也揭示了为什么它们对于将来的移动服务是必需的。本书向读者展现了最新的对于上下文智能的挑战和机遇，解释了面向上下文编程以及在服务和上下文上以用户为中心的视角的潜在性，包括诸如智能用户轮廓、主动服务发现、主动选择等热门的话题。本书还适当介绍了各种技术并且展示了它们是如何有机地结合在一起，从而更好地满足用户体验的需求的。

本书读者范围

本书可以作为决策者、商业战略家、首席技术官（CTO）以及电信管理者等对未来服务机遇的潜在性进行评估的有益参考。同时，本书也适用于研发领域的电信工程师，适合于那些在计算机、电子和电气以及电信领域对下一代移动信息技术和服务具有深刻理解和认识的科研人员。本书还适用于大专院校相关专业的学生，它为他们提供了宝贵的资料，使他们能够深入的理解服务支撑系统。

本书的组织结构

通过序和前言的导引之后，本书便开始了向读者展示未来服务供给的历程。正文部分谈及了服务支撑机制的许多方面，包括上下文相关和个性化。它描述了复杂服务蓝图的问题范围以及邀请读者通过“移动探险”来对它们进行探索。

本书的3个主要部分安排如下：第一部分首先介绍了新一代服务体系架构的主要支柱。它从构建移动通信网络开始，并且考虑到连接技术的多样性以及网络拓扑的多样性。接下来的章节就是对服务平台的透彻描述，主要通过聚焦于移动服务支撑来表现它们的功能。第一部分中还介绍了下一代服务体系架构以及其向普适化的扩充，紧接着介绍的内容是点对点系统以及移动中间件的相关问题。最后，总结了基于跨层设计的系统优化方法。第二部分重点介绍了智能服务提供的基础。应用本体论技术以及描述逻辑来增加语义服务的可用性的益处是讨论的核心内容。这一部分总结了整个系统的动态适应技术，从而描述了形成一个开放的适应性软件基础的功能组件。第三部分则重点强调了服务在环境中的智能潜入问题。它开始讲述了上下文感知的移动管理，接着描述了面向上下文的编程原则、主动服务发现、选择机制，以及智能服务组成。本部分以一个使能者的角度总结了现代移动周期，然后总结了个人移动和移动个性化的问题。最后以展望一个拥抱提供主动服务的真实世界作为本书结尾。

致谢

这本书的创作灵感来自于德国慕尼黑DoCoMo欧洲通信试验室的下一代网络实验室多年的研究经验。许多来自于这个实验室研究人员的观点和想法对本书的形成都起到了至关重要的作用。很自然地，他们的想法也贯穿于这本书当中。并且，在日本横须贺与DoCoMo研究实验室的合作过程中，他们也给出了很多有益的建议，包括我们的研究方向及我们在本书中所描述的下一代通信服务提供的潜在主题等。同时，本书的许多观点也来自于我们与欧洲许多大学、研究机构的合

作，是他们激发并促成了这本书的诞生。

在此感谢上面提到的所有对本书有贡献的人。最后，我要对 Wiley 出版社表示诚挚的感谢，感谢他们在本书撰写和出版过程中给予的技术支持。没有他们一贯的鼓励和支持，也不会有这本书的出版。

Hendrik Berndt

德国　慕尼黑

关 于 作 者

本书的所有作者都是 DoCoMo 未来网络实验室的成员。他们具有丰富的著书经验，并且都拥有很多项专利，其中很多人还曾为一些著名服务研究团队以及网络和软件技术方面的学术会议的程序委员会成员。

Christian Bettstetter：奥地利 Klagenfurt 大学网络和嵌入式系统中心的主任和移动系统方面的教授。他的研究领域包括网络、算法、协议以及无线网络建模。在过去的 8 年中，他的主要研究领域是多跳网络。在评为教授之前，他是未来网络实验室的高级研究员。

Zoran Despotovic：DoCoMo 欧洲通信实验室普适网络研究组织的高级研究员。他的主要研究领域为分布式系统，尤其是 P2P 系统。他从贝尔格莱德大学获得了工学硕士学位，从瑞士联邦技术学院获得了计算机科学的博士学位。

Robert Hirschfeld：Potsdam 大学 Hasso-Plattner 学院的计算机科学专业教授。他创立并领导了一个软件体系架构的团队，主要进行改善大型复杂系统的体系和设计方面的软件工具的研发工作。他也是 DoCoMo 欧洲通信实验室的高级研究员，主要工作内容是研究下一代移动通信系统的架构和功能组件，并关注与动态服务的自适应以及面向上下文的编程等问题。他从德国的 Ilmenau 技术大学获得了计算机科学专业博士学位。

Wolfgang Kellerer：DoCoMo 欧洲通信实验室普适网络研究组织的高级管理员。他的研究兴趣包括移动服务平台、P2P 网络、传感器网络以及跨层优化。他分别于 1995 和 2002 年从德国慕尼黑 Technische 大学获得硕士和博士学位。

Marko Luther：DoCoMo 欧洲通信实验室智能和安全服务研究团队的高级研究员。他是形式化方法、本体论技术和上下文感知的移动应用方面的专家。他获得了计算机科学专业硕士和博士学位。

Chie Noda：日本 NTT DoCoMo 公司通信器材开发部门的助理经理。她参与了多项电信标准化活动和欧洲研究项目。她从 2001 ~ 2006 年出任 DoCoMo 欧洲通

信实验室未来网络实验室的高级研究员。她获得了数学专业硕士学位。

Massimo Paolucci：DoCoMo 欧洲通信实验室智能和安全服务研究团队的高级研究员。他是服务语义呈现方面的专家，他目前的工作关注于基于短距离通信的移动应用。他从 Milan 大学获得计算机科学专业硕士学位。

Christian Prehofer：诺基亚研究中心的研究团队的领导者。他的研究兴趣包括自组织和普适系统，软件体系架构以及移动通信系统方面的软件技术。他从 2002～2006 年担任 DoCoMo 欧洲通信实验室未来网络实验室的项目经理。他分别于 1995 年和 2000 年在慕尼黑 TU 大学获得博士和 habilitation 学位。

Marco Sgroi：无线传感器网络实验室的主管，该实验室坐落于美国加利福尼亚州的伯克利，由 Pirelli 公司和意大利电信公司资助成立。他的研究领域包括基于平台的面向通信网络和传感器网络新的应用场景的设计方法学。他在 2002 年从加利福尼亚大学伯克利分校获得了电气工程和计算机科学的博士学位。在 2004～2005 年他出任 DoCoMo 欧洲通信实验室未来网络实验室的高级研究员。

Matthias Wagner：DoCoMo 欧洲通信实验室智能和安全服务研究团队的高级经理。他的专业领域包括主动服务、上下文感知以及移动应用中的语义 web。他获得了计算机科学专业的硕士和博士学位。

Jörg Widmer：DoCoMo 欧洲通信实验室普适网络研究组织的项目经理。他的研究领域是 MAC 层设计、网络编码、无线多跳网络的算法研究和未来网络结构。他分别于 2000 年和 2003 年在德国的 Mannheim 大学获得了计算机科学专业硕士和博士学位。

目　　录

译者的话

序一
序二
前言
关于作者

第 1 章　4G 移动新架构 …… 1
1.1　以用户为中心的时代 …… 1
1.2　商业模型上的考虑——开放编程环境的需求 …… 3
1.3　泛在服务和网络环境 …… 5
1.4　上下文感知 …… 9
1.5　个性化 …… 11
1.6　未来的移动服务 …… 12

第 1 部分　新体系架构的核心支撑

第 2 章　移动通信网络 …… 17
2.1　无线技术概览 …… 17
2.1.1　分类 …… 17
2.1.2　WLAN：IEEE 802.11 …… 21
2.1.3　蜂窝网络：GSM 和 UMTS …… 23
2.2　无线网络一览 …… 26
2.2.1　蜂窝网络的架构 …… 27
2.2.2　移动性管理和切换 …… 28
2.2.3　寻呼 …… 29
2.2.4　漫游 …… 29
2.2.5　服务质量 …… 30
2.2.6　位置服务 …… 31
2.2.7　广播和组播服务 …… 31

2.3　基于IP的下一代移动网络 …… 32
2.3.1　异构接入和移动性 …… 33
2.3.2　IP服务质量 …… 35
2.4　泛在计算和自组织组网 …… 38
2.4.1　移动自组织计算 …… 39
2.4.2　多跳无线接入网和网状网 …… 41
2.4.3　传感器网络 …… 41
2.4.4　可穿戴计算 …… 43
2.4.5　车载网络组网 …… 44
2.4.6　周边环境组网 …… 44
2.5　可编程网络 …… 45
2.5.1　适应性的概念 …… 45
2.5.2　可编程网络基础设施节点 …… 46
2.5.3　可编程的移动终端架构 …… 47
2.6　小结 …… 48

第3章　移动服务系统 …… 49
3.1　服务平台一览 …… 49
3.1.1　移动服务和支撑平台 …… 50
3.1.2　电信服务平台 …… 51
3.1.3　蜂窝网络服务平台 …… 55
3.1.4　基于IP的移动服务平台 …… 57
3.1.5　开放API …… 62
3.1.6　移动互联网 …… 63
3.1.7　移动服务平台需求 …… 67
3.2　下一代服务体系架构 …… 68
3.2.1　挑战 …… 68
3.2.2　通用服务特征 …… 70
3.2.3　泛在服务特征 …… 71
3.3　泛在服务示例：会话移动性 …… 71
3.3.1　移动性分类 …… 72
3.3.2　支持SIP的泛在服务环境中的服务移动性 …… 74
3.3.3　实现会话移动性 …… 76
3.3.4　发现泛在设备 …… 78
3.4　小结 …… 79

第4章 泛在性扩展：移动P2P …… 81
4.1 P2P系统的定义和分类 …… 82
4.1.1 非结构化P2P网络 …… 84
4.1.2 结构化P2P网络——DHT …… 85
4.2 DHT的一些问题 …… 88
4.2.1 维护开销 …… 88
4.2.2 复杂查询 …… 89
4.3 移动P2P …… 90
4.3.1 P2P技术给移动用户、运营商和服务提供者带来的好处 …… 91
4.3.2 移动P2P应用 …… 91
4.3.3 移动P2P面临的挑战 …… 93
4.3.4 异构移动环境下的移动P2P叠加网络 …… 94
4.3.5 P2P叠加网络和物理网络感知性质 …… 96
4.3.6 P2P信任和信誉度管理 …… 97
4.4 小结 …… 98

第5章 移动中间件 …… 99
5.1 移动中间件技术 …… 100
5.1.1 订阅/发布中间件 …… 101
5.1.2 反射中间件 …… 101
5.1.3 移动代理中间件 …… 101
5.1.4 P2P中间件 …… 102
5.1.5 移动中间件平台的挑战 …… 102
5.2 结构组件 …… 103
5.2.1 服务支撑层 …… 103
5.3 动态服务交付模式 …… 105
5.4 智能设备支持移动中间件 …… 107
5.4.1 智能卡技术 …… 108
5.4.2 无签约智能卡技术 …… 109
5.4.3 RFID技术 …… 110
5.4.4 智能设备举例 …… 111
5.5 小结 …… 111

第6章 跨层设计——一种新的移动通信系统优化方法 …… 113
6.1 简介 …… 113

6.2 跨层功能架构 …… 116
6.3 跨层优化的实现 …… 118
6.4 视频流媒体系统中的跨层优化 …… 120
6.5 无线视频流媒体跨层优化架构 …… 121
6.5.1 抽象层参数 …… 122
6.5.2 优化 …… 123
6.5.3 性能和成本分析 …… 124
6.6 小结 …… 126

第2部分 基础智能服务的提供

第7章 本体 …… 129
7.1 描述逻辑 …… 129
7.1.1 基础描述语言 …… 130
7.1.2 推理服务 …… 131
7.1.3 语言扩展 …… 133
7.1.4 描述逻辑系统 …… 135
7.1.5 应用 …… 136
7.2 Web本体语言 …… 136
7.2.1 语言元素 …… 137
7.2.2 子语言 …… 140
7.2.3 基于规则的扩展 …… 140
7.2.4 语言缺陷 …… 143
7.3 本体工程 …… 144
7.3.1 设计原则 …… 144
7.3.2 结构化 …… 145
7.3.3 开发流程 …… 146
7.3.4 标准本体 …… 147
7.3.5 开发环境 …… 149
7.4 小结 …… 150

第8章 语义服务 …… 151
8.1 挑战与机遇 …… 151
8.2 实际体验 …… 152
8.3 Web服务 …… 154
8.3.1 Web服务描述语言 …… 154

8.3.2 统一发现、描述与集成规范 …… 155
8.3.3 面向服务的体系架构 …… 156
8.4 从XML到本体 …… 157
8.5 使用语义网技术表示服务 …… 159
8.5.1 服务配置 …… 161
8.5.2 基础设施无关性 …… 164
8.6 过程模型和绑定 …… 165
8.6.1 使用OWL-S过程模型与服务交互 …… 167
8.6.2 使用OWL-S查找服务并与服务交互 …… 167
8.7 关于Web服务语义的其他方案 …… 168
8.7.1 Web服务建模本体 …… 168
8.7.2 WSDL-S和SAWSDL …… 169
8.8 语义在Web服务中的应用 …… 170
8.8.1 使用语义服务方便用户的交互 …… 170
8.8.2 使用用户上下文和偏好进行计算 …… 171
8.8.3 使用语义控制能量消耗 …… 172
8.9 问题和未来的挑战 …… 172
8.9.1 循环中的用户 …… 173
8.9.2 信任和隐私 …… 173
8.9.3 完成循环 …… 173
8.9.4 使用上下文信息 …… 174
8.9.5 Web服务组合 …… 174
8.10 小结 …… 175

第9章 动态适配——实时调整服务 …… 177
9.1 导言 …… 177
9.2 方法 …… 178
9.2.1 模块化和变更点 …… 179
9.2.2 面向方面的编程 …… 179
9.2.3 后绑定与反射 …… 180
9.2.4 平台 …… 181
9.3 案例：第三方服务集成 …… 182
9.3.1 基础服务 …… 183
9.3.2 辅助安全措施 …… 183
9.3.3 设计指南的一致性 …… 185

9.3.4　后用户界面的品牌化 …… 187
9.3.5　升级、更新和补丁 …… 189
9.3.6　服务集成 …… 190
9.3.7　使用说明和计量 …… 191
9.4　展望——面向上下文的编程 …… 193
9.5　小结 …… 194

第3部分　环境中的服务和智能嵌入

第10章　上下文感知的移动性管理 …… 197
10.1　上下文感知的移动性管理实例 …… 198
10.1.1　上下文感知切换 …… 198
10.1.2　定制寻呼服务 …… 199
10.2　移动网络中的上下文管理 …… 201
10.2.1　上下文管理和上下文感知切换的相关工作 …… 202
10.3　上下文感知切换的上下文管理方法 …… 202
10.3.1　上下文交互协议方法 …… 203
10.3.2　上下文感知决策代理的动态下载 …… 205
10.3.3　综合方案 …… 207
10.4　移动网络上下文管理的体系架构 …… 208
10.4.1　移动网络中管理上下文信息的体系架构 …… 210
10.5　上下文感知移动性管理的实现 …… 211
10.5.1　动态节点平台和服务配置 …… 212
10.5.2　节点服务配置 …… 213
10.5.3　上下文感知切换的网络层服务配置方案 …… 213
10.5.4　整体流程 …… 213
10.5.5　评估场景 …… 214
10.5.6　实验和评估 …… 214
10.6　小结 …… 215

第11章　智能上下文 …… 217
11.1　简介 …… 217
11.2　研究原型 …… 217
11.2.1　OWL-SF …… 218
11.2.2　上下文感知器 …… 220
11.2.3　McAnt …… 224

11.2.4 MobOWLS …… 226
11.2.5 PERCI …… 226
11.3 小结和展望 …… 226

第12章 从个人移动性到移动个性化 …… 228
12.1 简介 …… 228
12.2 未来用户配置和个性化 …… 229
12.3 新移动生活 …… 231
12.3.1 高级个性化概念 …… 231
12.3.2 上下文感知计算和管理 …… 231
12.3.3 灵活的服务支撑中间件及其演进 …… 232
12.4 主动服务发现的偏好模式 …… 233
12.4.1 具体应用场景 …… 233
12.4.2 用户偏好 …… 233
12.4.3 协同服务发现 …… 234
12.4.4 MobiXpl——面向基于偏好发现的用户界面 …… 235
12.5 面向移动个性化 …… 235
12.6 结论和展望 …… 237

第13章 主动服务走向现实 …… 238

附录 参考文献及延伸阅读 …… 241

第 1 章　4G 移动新架构

Hendrik Berndt

1.1　以用户为中心的时代

用户并不关心技术，用户需要的是能够满足他们需求的服务且关心该服务的价格，伴随着有趣且令人激动的技术的不断涌现。移动服务发展的趋势是能够让用户享用更多的服务，并且服务的使用更加容易。未来的服务应该以一种更为巧妙和顺畅的方式提供给用户，而且应该足够便宜，便宜到用户在使用时不用过多地考虑其价格。服务应该是随手可得的，可以通过智能的服务“向导”来引导用户，轻轻松松地在服务空间中畅游。信息爆炸和过量常常被用户所抱怨，但在未来不应该成为一个障碍。在未来的服务环境中，应该是以用户为中心的。

图 1-1 展示了从第 1 代移动通信到第 4 代移动通信的演进路线和它们的主要特点。这个图可以作为我们讨论的起点。从图 1-1 中可以看出，各代移动通信系统的区分主要是技术驱动的。每一代系统的生命周期都将近十年，其中包括对技术进行规范所需的研发时间和从开发部署到投入商用的时间。尽管在今天，无线应用和服务带来了无数的商业上的成功，包括信息数据服务，商业应用（如移

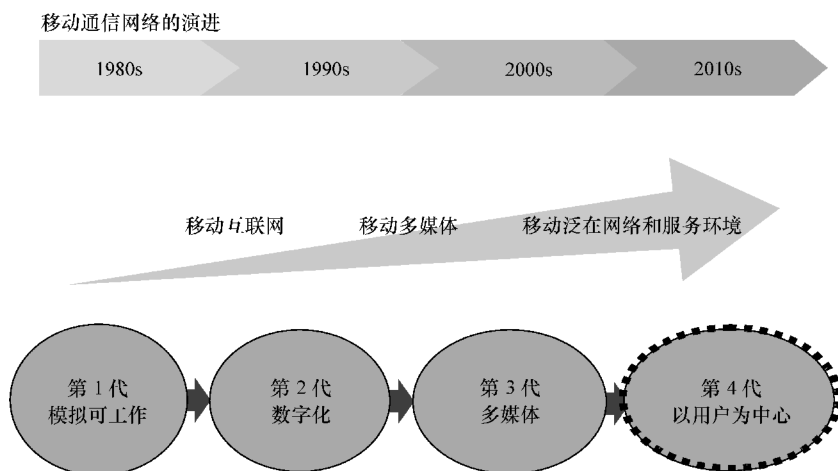


图 1-1　不同年代中无线通信的演进

动银行、移动支付等)，娱乐应用（如卡拉OK、在线游戏、音/视频流媒体等)，企业服务（如办公文档应用、局域网应用)，通信服务（如传统语音、短消息、电子邮件）等，但只有在未来3G以后的系统中，用户才能真正成为服务的驾驭者，主导服务不断的发展。

那么，我们写这本书的时候，是站在什么地方呢？

基于IMT-2000的3G移动服务同时提供传统的语音服务和高速数据服务，如电子邮件、即时消息、多媒体和互联网接入，实现了传统的移动网络和宽带网络的融合。今天的移动宽带所带来的利润已经显示出了巨大的市场潜力。

以世界领先的移动通信运营商之一NTT DoCoMo为例，截止2007年9月，DoCoMo的3G服务系统FOMA™为4000万用户提供了服务。FOMA™可以说是世界上第一个3G服务系统，它于2001年启动。DoCoMo还提供了很多先进的移动多媒体服务，包括世界上最流行的移动电子邮件/互联网服务i-mode™，再加上信用卡以及其他一些电子钱包的功能，DoCoMo移动电话成为日常生活中功能丰富、必不可少的小工具，在日本i-mode™就它拥有4700万用户（参见http：//www. nettdocomo. com)。

依照图1-1的逻辑，后3G的全球研究应该在刚进入2000年时就开始了，事实情况确实如此，新涌现的解决方案包括超级3G、3G长期演进、IMT-Advanced和4G等，这些工作主要致力于提出新的空中接口技术。最有潜力的技术可能是基于多载波传输的技术，它能够在广域的环境下提供100Mbit/s的速率而在热点地区可提供1Gbit/s的速率。需要注意的是，作为系统部署的一个必要条件，应该为下一代移动系统分配足够的频带资源。

对于服务而言，通向4G之路就不那么明确。在未来，一个单一的服务将不能满足用户的需求并获得用户的忠诚度。服务应当被嵌入到一个支持用户发现服务，并在运行时根据运行条件指导服务使用的框架中。服务的选择应该基于用户的偏好，而且能基于上下文感知的服务交付机制进行运行时的适配。一旦这些理念得到实现，我们将拥有一个全新的服务环境。

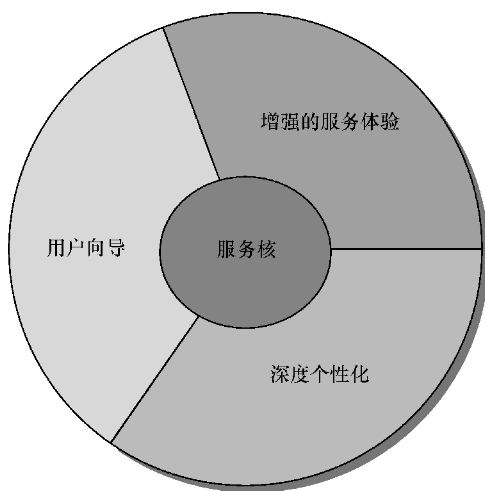


图1-2　通用服务提供模型

通用服务提供模型是新一代的TINA USCM，它提供了这些针对未来服务交付方式的增强原则，如图

1-2 所示。核心部分表示服务的首要特征，经过适配后，服务可以以个性化实例的形式来交付，并能够感知周围的状态。同时，与服务共同交付的还包括一个服务指南，便于用户理解服务。为了增强服务体验，可以增加一些虚拟现实的元素。人的感知以前还没有作为通信的一个部分，现在可以将这个特性作为“Service Universe”的一个部分。未来的一些扩展能够使服务主动就绪和执行，在本书中使用“主动服务”来对这些特性加以解释和定义。

本书的目的是为了阐述新架构的方方面面，并为新的服务特性做好准备。相比于现在的服务，这些服务在语义上得到了增强，根据客户的情况和偏好主动应对，提高用户的满意度。

无所不在的连接，以及由成熟的服务平台和能够基于上下文智能采取动作的服务逻辑所支撑的服务交付机制是新一代服务最重要的构成部分。

1.2　商业模型上的考虑——开放编程环境的需求

传统电信服务的商业模型只涉及两方，形成了一个垂直的封闭服务模型，即提供者/用户模型。在这个模型中，电信运营商既提供网络基础设施也提供服务本身。增强的服务特征会随着时间的推移不断出现，但这个商业模式的核心仍然是端到端的、封闭的、私有的解决方案，将用户绑定到运营商的服务之上。

这种封闭的垂直服务架构要求电信运营商维护所有的基础设施和服务，因此一些很有希望的第三方服务和解决方案通常会被排除在外。

在实现更好的开放性方面，目前已经取得了一些进步。内容和应用提供商可以通过开放 API 向客户提供服务，这些服务通过与电信运营商提供的计费、安全等特性的集成在功能上得到了增强。特别是越来越多的移动运营商采用了被称为“半围墙花园”的商业模式，用户可以从移动运营商认可的、签有合作协议的第三方获取内容和服务，但同时也并不限制用户调用其他任何第三方服务运营商所提供的服务。

然而新一代的移动互联网的成功还需要依赖于新的商业模式。这种商业模式将从上面所描述的模式演进，在开放性和适应性上达到一个新的水平。新的商业模式应能允许各种新进入者参与进来，与新的商业角色进行合作，重新定义运营商、服务、带宽代理、应用提供商、签约用户之间的角色，并为所有的投资者带来新的收益。

这样一个商业模式需要构建在一个开放的、具有适应性的、可编程的系统架构之上，其具有以下特性：

1）通过明确的参考点和开放的 APIs，边界应定义清晰。

2）各部件之间应提供良好的互操作性，并确定各参与方的商业角色和确保

服务的提供。

3）系统各个部件有各自独立的演进周期，能够快速部署、快速响应。

在以往的大型移动通信系统设计中，有很多相互独立或相互依存的部件，它们以分层的方式进行设计。在架构中层的使用是基于水平化的结构，就像一个分层的蛋糕，垂直的各个栈与层中的水平操作是相互正交的。分层的模式支持垂直信息在高层和底层之间的双向流动。

4G系统从无线传输到应用都需要根据快速变化的环境进行调整，因此，必须提高演进的和可适应性的能力。其目标是提供动态的适应性机制，这种机制要应对来源于多用户、多设备、多平台、多服务和上下文等因素的复杂环境。适应性使得应用层能够基于信息的变化（用户信息描述文件、偏好、上下文信息）做出响应，这些信息来自于个性化的设置和对上下文的感知。适应性还应该能对底层协议的变化做出响应，包括可用的数据传输速率、当前的信道条件等，典型的情况包括连接特性的变化、用户进入到一个新的服务区域、在服务会话中终端的变更等。相应地，我们可以将适应性分为：

1）媒体适应性，如文本到语音的转换。

2）内容适应性，如呈现或信息的排序等。

3）影响服务逻辑的服务行为适应性。

适应性必须既有被动的特征又有主动的特征。适应性既能够让用户主动请求，如将文本转换成语音，又能够基于上下文信息自动执行，如在移动电话上变更信息的呈现形式。

为了实现适应性的通用解决方案，我们提出了由抽象层构成的架构，每一个抽象层由一个或多个开放平台组成。因此，我们就不仅仅要支持层次之间的开放接口了。图1-3描述了开放平台的概念，其中平台是由基础平台和平台组件组成的，平台组件经过修订能够根据不同的情况来适应不同的需求。

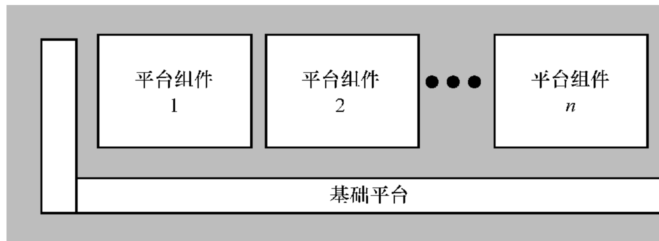


图1-3 由基础平台和平台组件构成的开放平台

（Courtesy of Journal of Communications and Network 授权）

在图1-4中，具有跨层协作能力的开放式的可编程架构作为服务和中间件平

台，其组成部件和接口必须调整形成一个适应性单元。

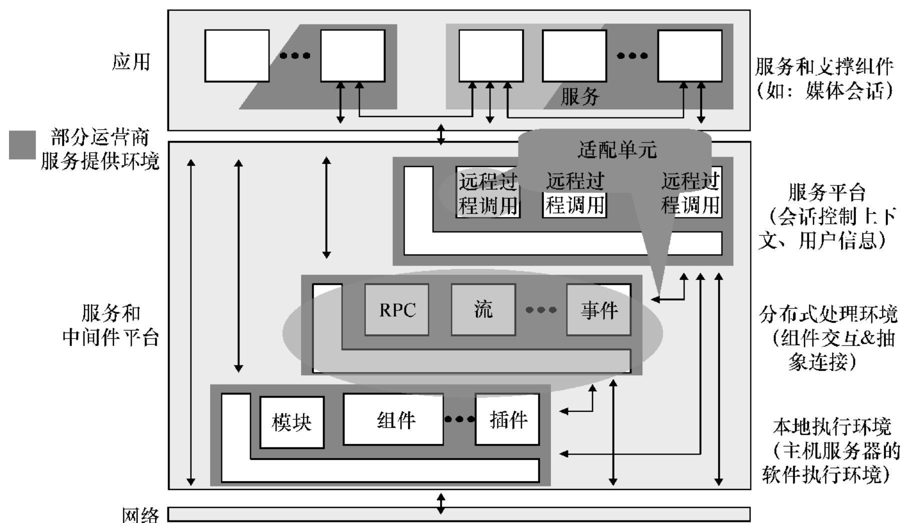


图 1-4　跨层协作的配置服务和中间件架构

这样一个开放式的可编程架构（见 2.5 节）及其内置的跨层协作能力将减轻新服务的引入或基于现有系统引入新的服务特性时的负担。随着适应性单元的引入和可变点的引入能够将系统的灵活性和适应性带到一个新的高度，并能够处理一些未曾预料到的系统升级的问题，这些将有利于满足 4G 移动通信系统的需求，其中可变点是指能对软件组件进行高效修改的点。具备了增强的泛在计算平台、基于上下文的服务调整和自适应的服务后，主动服务将成为现实。

1.3　泛在服务和网络环境

随着数字电子产品的发展，计算机变得越来越小，且越来越便宜。同时，处理能力和存储能力则以令人惊叹的速度在提高，这两种趋势的组合使得计算能力进入到了以前的技术不可及的位置和设备中。而且，越来越多的日常用品中也植入了无线接口，如图 1-5 所示，它们之间可以相互联网。一些新的、有趣的商机涌现出来，日常生活、病患护理、机器人、后勤等都可能成为应用领域。在不久的将来，人们将被更多的计算机所包围，看起来我们就像居住在计算机的国度中一样。这种场景被称为泛在服务和组网环境，这是一种由计算资源和微小的计算设备所环绕的执行环境。

要营造这样的环境在技术上户面临一些挑战，本书中的一些章节将对这些问题进行描述。在这个介绍里面我们可以先谈及一二。首先，需要动态配置无数的

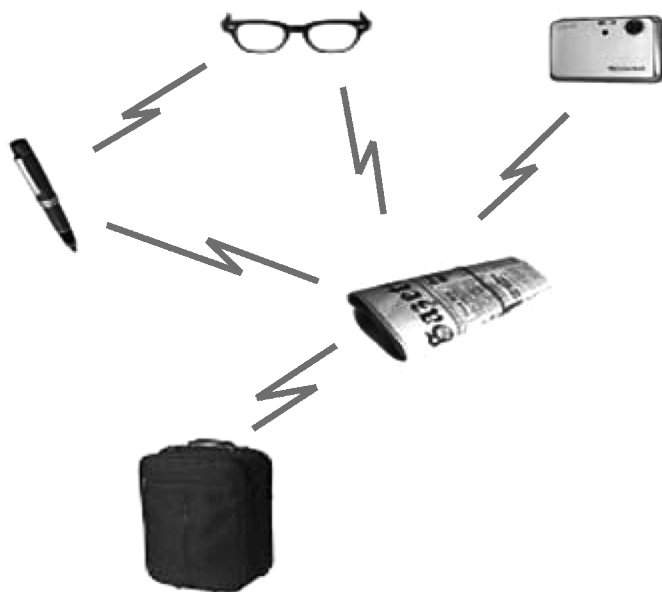


图1-5 将很快拥有无线接口的日常用品

网元，才能构建出合适的拓扑结构，这些拓扑结构通常是和应用相关的。通过对这些网元进行控制和维护，从而能够在上面部署新的应用。要管理这样一个泛在的环境，用集中式的控制方式远远达不到要求。可以类比于自然界的情况，自然界更偏向于自组织的解决方案。自组织是能够用简单的规则来完成复杂的全局过程的关键，一些形成自组织系统的基本原则如图1-6所示。

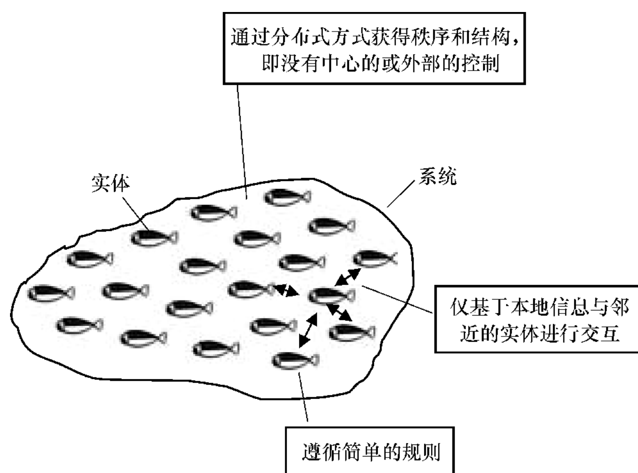


图1-6 自组织的优势：系统行为仅由与邻近的节点进行简单的交互而形成

一个具有自我动作的行为，包括自组织、自愈、自配置等，是构成自适应性、全局可访问、易配置和能够被公平使用的网络的要素。但它们一起所产生的复杂性要远高于每一个单个网元。鲁棒性也是通过大量的节点来获得的，在这种情况下，一个节点失效并不会影响整个系统的可靠性。

第二个挑战是网络中的网元存在着很大的差异，如从 RFID 标签、靠电池供电的传感器和具有很短生存期的设备到可以使用交流电源，从而能保证一直在线的设备。泛在网络环境的一个重要特征就是网元经常受到电源的限制。这意味着与基站之间的远距离连接并不是一直可用的，因此使用多跳通信的场景是非常有意义的，一些网元需要引入中继的功能。

第三个挑战是泛在环境中的组网和应用是高度相互依赖的或集成在一起的，这点不同于很多其他的通信系统。传感器通常被设计为仅用于一种单一的应用。目前，一些新的技术可以克服这个缺陷，对特定网络的功能进行“调频”或引入一些诸如“模式切换”的机制，从而使得传感器网络能够根据变化的环境改变拓扑。

第四，尽管网络技术和服务都存在多样性，下一代移动通信网络还是需要为用户提供无缝的接入。各种各样的接入技术和接入网络可以使用户随意漫游于各种网络之间（见 2. 3. 1 节）。对于服务提供而言，无缝的接入意味着用户能在任何地方、任何时刻、任何方式访问到其服务和用户信息。这需要在网络层和应用层引入新的移动性的概念。在 3. 3. 1 节中，我们将讨论不同的移动性，包括终端移动性、个人移动性、服务移动性和会话移动性。

最后，当所有的人和所有的物体都互连起来时，安全和隐私问题就变得很重要了。作为一种扩展的功能，在泛在服务和组网环境中的个人设备可以提供认证和授权的功能，不仅能够用于自身，还可以用于其他移动或非移动的服务，如可以利用近距离通信来调用火车票订购、企业网络访问等服务。

有很多的方法可以用于对抗安全威胁和隐私侵犯，如可以使用移动运营商仲裁的方法，在泛在的环境中压制恶意节点的行为。在该场景下，一些由运营商提供的设备，如移动电话或个人数字助理等成为增强如自组织网络或其他泛在网络网元能力的关键部件。之前已经建立好客户-运营商关系的设备可以看成是一个由信任节点组成的叠加网，如图 1-7 所示。

运营商部署的节点通过多条路径来交换路由信息，从而能够检测出信息中的异常。由此，它们具有更加有效方法来检测恶意节点，并因为它们了解网络的总体情况，所以具有重新路由的能力，如图 1-8 所示。

总结这个对于泛在服务和组网环境的介绍部分，我们强调了未来移动系统必须面对连接性和服务提供可能会以无法预测的规模发展的挑战。包括人和人的通信以及机器和机器的通信，任何东西或人都有可能进入到用户的使用环境中。我

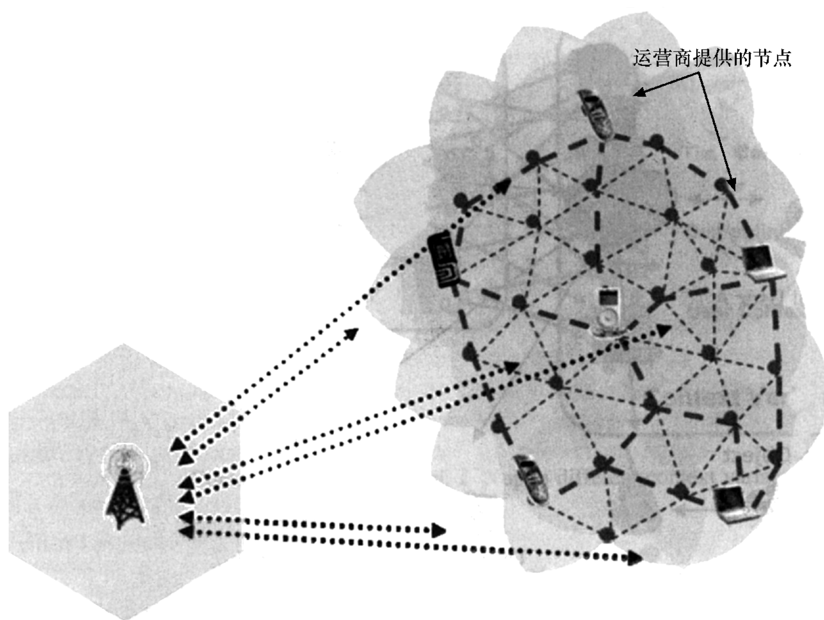


图1-7 运营商仲裁的通信

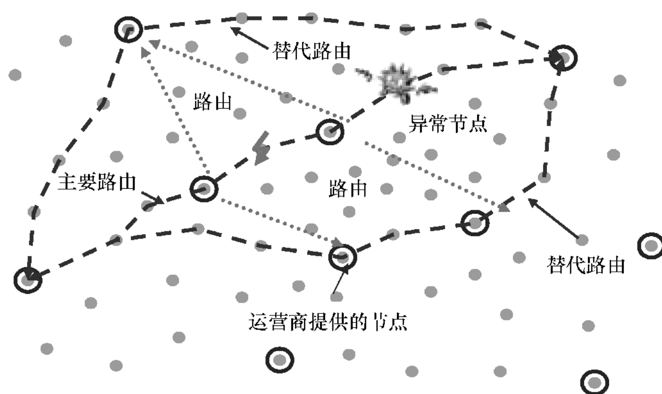


图1-8 恶意节点的检测以及通过网络提供商仲裁的重叠网节点来建立替代路由

们相信如移动运营商之类的投资方将会把泛在计算与蜂窝网络结构融合起来以获取新的商机。我们将以一个有价值的应用来结束本节的介绍。我们经常会忘记我们的随身物品放在什么地方，通过在有价值的东西上加上标签，并经由无线技术，如RFID、蓝牙或类似技术，就可以被传感器所感知到。移动网络中所使用的设备可以作为移动基础设施和自组织传感器网络之间的“泛在网关”。在寻找这些丢失物件时可以向一组远程的泛在网关发送查询消息，当在传感器感知的范

围内感知到所丢失的物件时，泛在网关会触发一个响应。图 1-9 描述了这一特定的应用情况。

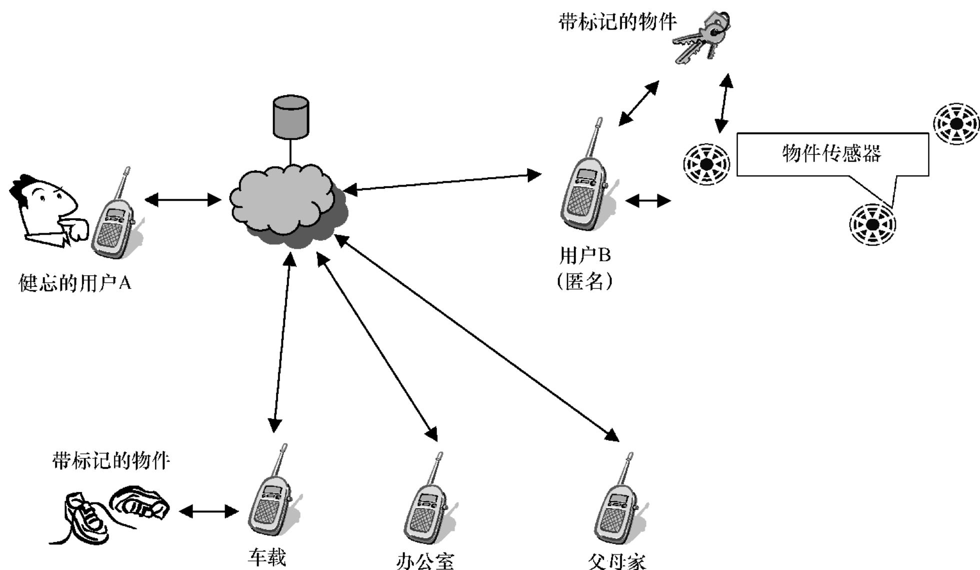


图 1-9 “找东西”一个通过移动网络，利用泛在服务和组网环境来找到错放的、丢失的或被盗的个人物品的例子

1.4　上下文感知

作为主动服务，展现移动应用的真正魅力是在提供服务时，能够感知并响应各种环境变量。环境变量通常被描述为上下文，包括在服务运行时的用户状态信息。对用户状态的评估可以以一个上下文的数据解释（如位置），也可以根据多个。上下文可分为但不局限于以下几类：计算上下文，如网络连接；时间上下文，如一天中的不同时间；物理上下文，如重量级别；应用和对象的关系；甚至用户的社会地位。上下文也可以包括用户与服务之间的交互（应用历史）和未来希望的关系。图 1-10 描述了为对于特定的用户或者用户群体提供服务时，利用不同的上下文建立特定的上下文空间。基于上下文空间的上下文感知服务的提供，允许为限定上下文界限的建立，在这些界限里存储和选择上下文的入口对象。即便是最复杂的上下文空间场景，也只能表示出整个状态变量集合的一部分视图。

举例来说，一个用户可以将应用从一个情形上下文迁移到另一个上。在这个新的上下文中将面临更加复杂的上下文环境（见图 1-11）。此外，环境属性的动

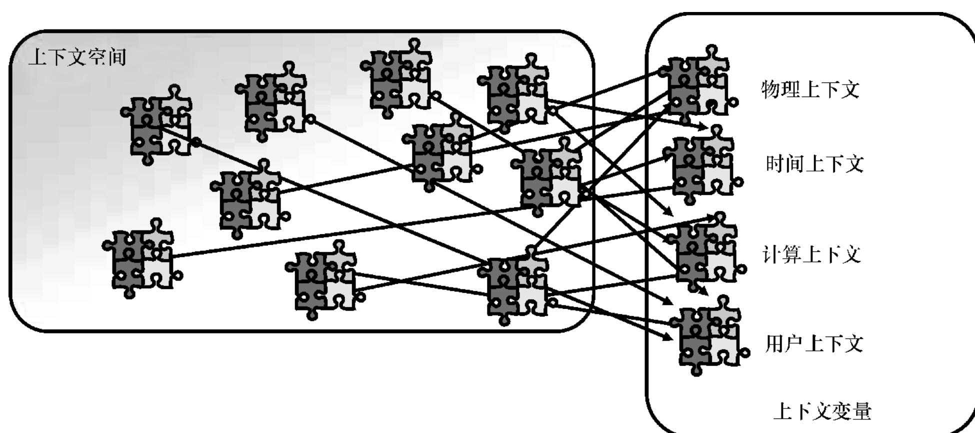


图1-10　通过上下文空间概念构造的上下文变量

态性、连续性以及不确定性，在一定程度上增加了了解环境的复杂程度。环境状态将会动态地改变，并且独立于服务执行。不存在一个离散的固定数值描述上下文，更进一步，无法保证可用的状态信息是精确的，服务对环境状态只有部分可控。基于这些对上下文的不同理解，主动服务与以前已有的服务不同，它必须应用于高度动态的场景里。

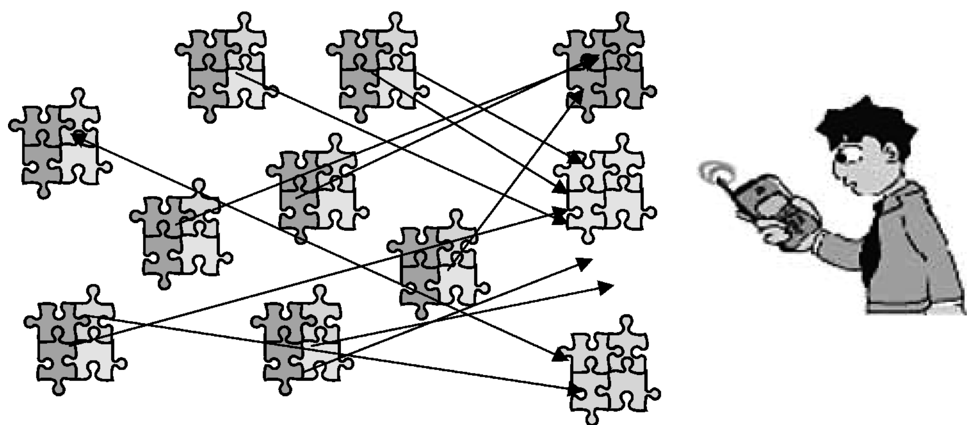


图1-11　用户和嵌入在复杂上下文环境中的服务

处理上文中描述的环境需要一些行之有效的手段，包括：

1）连续的和分布的上下文建模。

2）上下文搜索，信息聚合和抽象。

3）上下文分组和过滤。

此外，还需要：

1）自适应算法，能够从示例中搜集知识。

2）外界原因组成，解释附近的状态。

3）通用机制，将知识应用于新的环境。

本书的第 3 部分通过对主动服务和它们应对不同环境时的智能反应来回答上面的问题。

1.5　个性化

深度的个性化可以针对用户的偏好提供经过精简的服务，在给定的上下文中和特定的情况下自动地反映出用户的需求。理想的情况下，服务主动地启动并就绪，以便于在需求时能被立即使用。从本质上来说，深度的个性化远不止于利用个人数据来配置移动设备或根据设备能力如显示屏尺寸来对内容进行适配等。个性化致力于一个完整的用户模型，在提供服务时可用于所有的任务。最新的标准关注于用户的标识以及个人接入等，仅包括了最基本的用户模型和信息组织。这些标准化工作包括：

1）CC/PP（World Wide Web 协会组合能力/偏好信息 —— http：//www. w3c. org）。

2）在 Liberty 联盟项目（http：//www. projectliberty. org）中所讨论的 NET PASSPORT（http：//www. passport. net）信息组织。

3）3GPP（http：//www. 3gpp. org）和 Parlay 组织（www. parlay. org）建议的用户信息。

除用户模型和信息组织，在服务和基础设施的个性化方面还存在一些挑战。用户可能不仅拥有一个通信设备，而且可能同时使用多个设备，其中的一些，如公共的互联网终端，可能是从公共的环境中获取内容的。因此，网络中用户信息的可用性以及一些变更的同步都是必要的。信令机制和信息发现、查找的技术应能确保在正确的地方、正确的时间提供正确的信息。然而用户信息的交换会导致安全和隐私上的问题。服务提供商应能确保用户信息和数据的机密性和完整性。个性化服务和个人信息的披露之间的权衡是服务提供商和客户关系的关键点。客户关系和信任有关，信任不仅是安全的问题，还包括例如服务提供商是否能够从客户的利益出发，从而得到客户的信赖等。信任很大程度上是基于安全和隐私的，但同时也和声望有关。

到目前为止，我们都是从用户仅调用单个服务的角度来讨论个性化服务提供的。个性化的最终目标是要满足每个用户的需求，其中不乏一些复杂的任务。这些任务可以分解成多个子目标，这些子目标又可以匹配到不同的服务。以商务旅行为例，这个例子可以展示在用户的日常旅行中，泛在服务是如何贯穿的，以及一些相关的服务，如租车、寻找下一个 ATM 柜员机、预定一个便捷的酒店是如

何作为一个整体提供给用户的。这些服务，特别是通过手机进行访问时，需要一个全面的个性化支持。到目前为止，这些技术还不能完全支持。

随着服务的发展，服务多样性的呈现，需要引入一些以用户为中心的，基于用户偏好的服务发现和选择的技术。尽管现在已经有了 UDDI 和 WSDL 等用于实现服务目录，但它们还是缺乏对服务个性化概念的强力支撑，而这点对于用户使用是很重要的。为了解决这个问题，需要增加服务目录的能力，基于每个用户的特定需求和偏好，通过协作的数据库和过滤器来发现、选择和组合服务。我们关于个性化服务的观点如图 1-12 所示。用户信息，包括偏好等，是服务提供各个阶段的关键，服务的阶段包括：服务发布、服务发现、服务选择和服务执行。

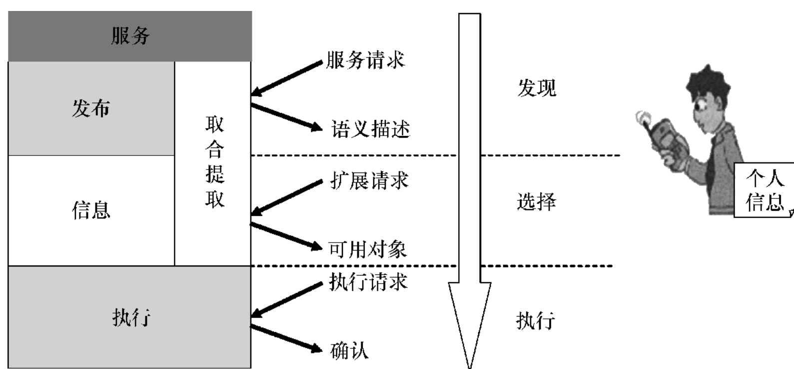


图 1-12　个性化的服务发现、选择和提供

后续我们会讨论到个性化的选择和执行。在这里我们只是简要地描述基于用户的特定需求和偏好的个性化发布和服务发现的简要步骤。通常使用模式或者信息内容更丰富的服务请求，加上显式或隐式给定的单个用户或用户组的偏好可以用于改进用户和服务之间的匹配。

1.6　未来的移动服务

未来的移动服务逐步朝着深度个性化、位置感知和主动服务提供等方向发展，在本书的后续章节将会对这几方面进行详细描述。随着技术的发展，增加对人意愿和需求的理解，对于通信产业来说具有越来越重要的意义，因为信息的过量已经渐渐成为常规而不是异常现象了。我们希望通过图 1-13 来举例说明未来的移动服务背景。

未来的移动服务以泛在网络为基础，包括形态各异的连接和通信信道、叠加网和底层网络等。无线通信是最具挑战性的一个，它包括通过中继增强的蜂窝通信、多跳接入、泛在网关功能，允许确保服务质量以及安全可靠的

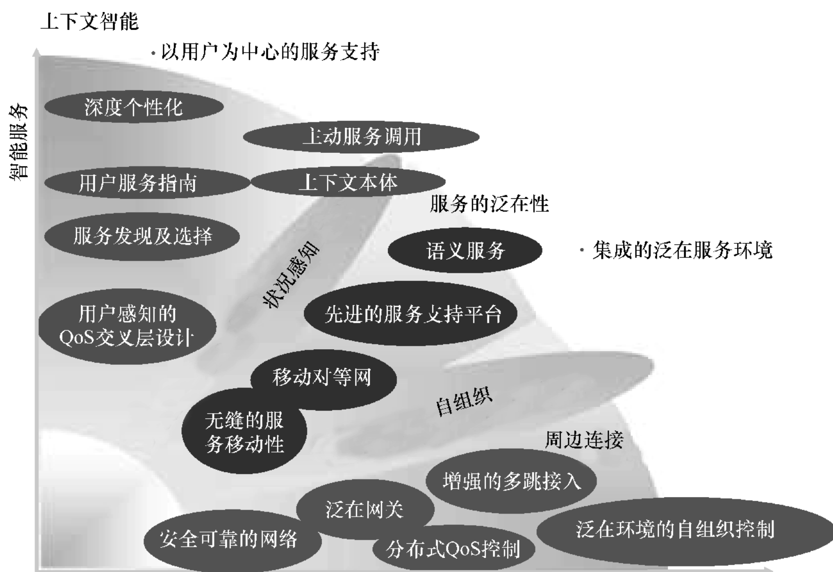


图 1-13　未来的移动服务背景

网络。未来的移动服务需要对网络连接进行有效的控制、管理和维护，这是需要引入新的组织方式的原因，最突出的组织方式就是自组织。附加的管理功能可以通过简化的管理叠加网来提供，这类叠加网应能够动态地去适配目前正在处理的任务[3]。

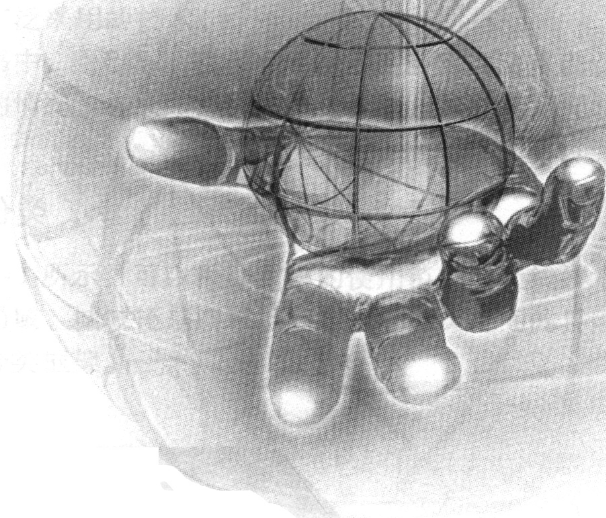
服务支撑平台提供了通用服务的特征，适用于许多服务和应用类型。通过运行这些服务，平台提供商可以更加了解平台使用的方式。服务支撑平台提供商可以增加元素来处理信誉管理，为客户关系建立可信的环境，并且采用灵活的计费和结算解决方案。服务支撑平台为无缝服务会话移动性提供了方法。该方法独立于内在的联系，并且能够为用户提供熟悉的接触和服务体验。语义增强服务可以通过更加复杂的方法自己表述自己，从而提供给用户友好的服务环境。

最后，我们希望上下文和位置感知智能化可以应用于服务发现、服务调用、服务执行中的每一步。该智能化着眼于服务交付，考虑了用户的资产和资源，用户的兴趣以及偏好，用户对需要的服务的熟悉程度，如果有必要还可以为服务整合用户向导。

未来的移动服务还可以进一步增强，如增加虚拟现实以及触及感官功能的元素，在本书结束部分的展望中我们还将简要地谈到这些未来的服务技术。

第1部分

新体系架构的核心支撑



第 2 章　移动通信网络

Christian Prehofer，Christian Bettstetter and Jörg Widmer

2.1　无线技术概览

移动服务是利用一些现代的技术，在无线频谱上实现的。本章对这些技术进行了一个简要的描述，并展望了一些即将出现的最新技术，让读者对这些技术的关键特性和应用范围有所了解。我们将聚焦于那些已经标准化并且在消费市场的产品中被广泛采用的技术。

在本书中，“无线”表示用户设备连接到网络基础设施或其他用户设备时不使用线缆的情况。在大多数情况下，无线连接都使用无线电波、光波或其他的传输介质。

2.1.1　分类

如图 2-1 所示，可以根据它们的使用范围对无线技术进行分类，即它们是用于个域、局域、城域还是广域网络。这些分类之间的界限并不严格，但提供了一个粗略的分类框架。

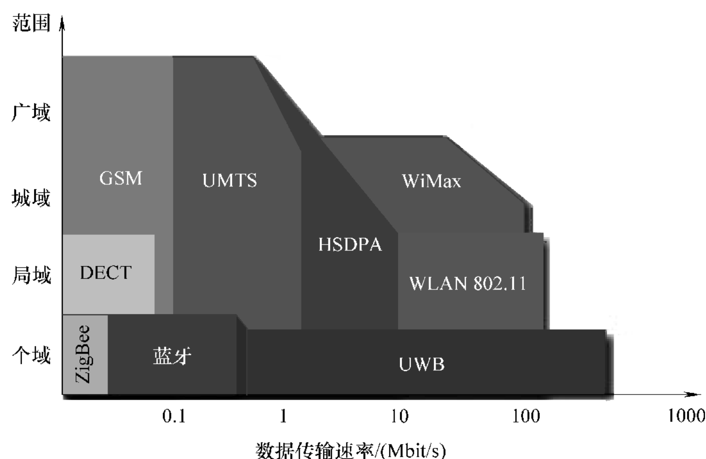


图 2-1　无线通信技术的分类

个域网是指个人所携带或穿戴的移动设备之间所形成的网络。一个典型的例子是音乐播放器或移动电话与头戴的耳机或显示设备之间所形成的网络（见图2-2）。如果我们展望更远的未来，个域网将互联可以穿戴在身上的计算机的各个组件。用于这个目的的无线技术主要是由美国电气和电子工程师协会（Institute of Electrical and Electronics Engineers，IEEE）开发的IEEE 802.15系列标准。这些标准包括的技术如ZigBee、蓝牙和超宽带技术（Ultra-Wide-Band，UWB）。其他用于个域网的无线通信技术包括红外传输（如IrDA标准）或通过皮肤传输的技术（如NTT RedTacton）。这些技术的作用范围通常是兆级，针对不同的应用，有不同的速率，最高可提供高于100Mbit/s的速率。

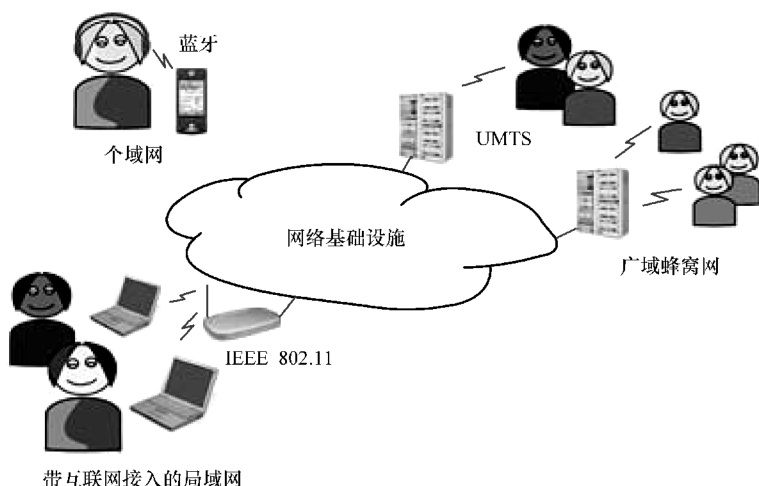


图2-2 无线通信技术的使用场景

局域网是指用于房子或办公室内的设备互联的网络，如固定或移动计算机、打印机、服务器、显示设备或电视设备之间的网络。目前最流行的无线局域网CWLAN产品都基于IEEE 802.11标准。该技术也称为WiFi，是过去若干年中出现的一个成功范例。WLAN节点被安装在校园、公司、机场、会议设施及其他一些公共场合，用户能够使用它们访问互联网、发送email等。随着宽带接入到家庭，住宅中的个人WLAN也越来越普遍。最新的IEEE 802.11产品提供数十兆的速率，能覆盖100m的范围（与环境有关）。另一种无线局域网技术是欧洲的数字增强无绳电信（Digital Enhanced Cordless Telecommunication，DECT）标准，这个标准用于千万家庭中的无绳电话。

城域网通常覆盖一个城市或一个城区。最新提出的WiMAX（IEEE 802.16）是无线城域网（WMAN）标准的例子。WiMAX在家中或办公室中可提供高速率

的 Internet 接入，并提供简单的移动性。WiMAX 收发器的有效距离是 10km 或更远距离，同时它还为家庭或办公用品提供了“最后一千米”的解决方案，能够将运营商网络连接到 WLAN 或直接连接到用户设备上。与优先接入网络，如光纤接入和 DSL 接入相比，WiMAX 的优点是部署成本更低。另一项 WMAN 的技术是 IEEE 802.20。

广域网通常的覆盖范围是一个国家甚至超越了国家的界限，它提供“任何时间、任何地点”的接入与访问。这种网络通常由一个公司或一个国家机构来运营，广域网的两个主要例子是全球移动通信系统（Global System for Mobile Communication，GSM）及其后继者通用移动通信系统（Universal Mobile Telecommunication System，UMTS）。用户与运营商签约后就可以用移动设备打电话、发短信、使用数据服务。UMTS 提供的数据传输速率为 100kbit/s 的量级。对于 UMTS 的扩展，高速下行分组接入（High Speed Downlink Packet Access，HSDPA）提供了传输速率为 10Mbit/s 的分组数据服务。另一个广域网的例子是数字音频和视频广播（Digital Audio and Video Broadcast，DAB 和 DVB）。能够覆盖全球的卫星通信系统，也属于这个类别之内。

表 2-1 ~ 表 2-3。总结了前述提到的技术的关键技术特征。下文将详细讨论更多的细节内容。

表 2-1　WPAN 无线技术中的关键技术特征

技　术	ZigBee（IEEE 802.15.4）	蓝牙（IEEE 802.15.1）	UWB
典型应用	小型设备和传感器组网	用于头戴式设备的线缆替代	多媒体数据同步，可穿戴计算
典型的范围/m	<50	10 或 100	4 或 10
数据传输速率	40kbit/s，250kbit/s	最大 1Mbit/s	>100Mbit/s
频率/GHz	0.868/0.915，2.4	2.4	3.1 ~ 10.6
物理层	DSSS	FHSS	脉冲无线电
MAC 层	CSMA/CA 和时槽分配	TDMA	TDMA
后向兼容性	N/A	N/A	N/A
标准化时间/年份	2004（2003：IEEE 802.15.4）	1998	正在进行
产品诞生年份	2004	2000	2007

表 2-2　IEEE 802.11 WLAN 无线技术中的关键技术特征

技　术	IEEE 802.11	IEEE 802.11b	IEEE 802.11a	IEEE 802.11g	IEEE 802.11n
典型应用	对互联网的无线接入	对互联网的无线接入	对互联网的无线接入	对互联网的无线接入	对互联网的无线接入

（续）

技　　术	IEEE 802.11	IEEE 802.11b	IEEE 802.11a	IEEE 802.11g	IEEE 802.11n
典型范围	室内30m左右，室外200m左右	室内30m左右，室外200m左右	室内15m左右，室外100m左右	室内30m左右，室外200m左右	室内100m左右
数据传输速率（Mbit/s）	最大2	最大11	最大54	最大54	>100
频率/GHz	2.4，无需牌照	2.4，无需牌照	5，需牌照和无需牌照两种	2.4，无需牌照	2.4，无需牌照
物理层	FHSS/DSSS	DSSS	OFDM	OFDM	多天线OFDM（MIMO）
MAC层	CSMA/CA	CSMA/CA	CSMA/CA	CSMA/CA	CSMA/CA
后向兼容性	N/A	IEEE 802.11	无	IEEE 802.11b	IEEE 802.11b
标准化时间年份	1997	1999	1999	2003	预计在2009
产品诞生年份	1998	1999	2002	2003	2006（标准前产品）

表2-3　DECT，蜂窝网络、WiMAX无线技术中的关键技术特征

技　　术	DECT	GSM/GPRS	UMTS/HSDPA	WiMAX（IEEE 802.16/IEEE 802.16e）
典型应用	家用无绳电话	移动电话，文本消息	移动电话，多媒体消息，互联网访问	无线互联网访问
典型范围	室内50m左右	城市中1～5km，最大30km	城市内数百米	3～5km
数据传输速率	32kbit/s	典型情况是30～70kbit/s，最大170kbit/s	典型情况是200～500kbit/s，最大2Mbit/s，HSDPA最大是10Mbit/s	每信道最大75Mbit/s
频率	1900MHz	900MHz，1800MHz，1900MHz，需牌照	2GHz，需牌照	IEEE 802.16：10～66GHz频带，需要在视距之内；IEEE 802.16a：2～11GHz频带，包括需牌照的和不需牌照的
多址方式	TDMA	TDMA	CDMA	10～66GHz：TDD和FDD的TDMA2～11GHz：OFDM
标准化时间年份	1992	1990	1999	2001/2005
产品诞生年份	1993	1991	2002	2006

2. 1. 2　WLAN：IEEE 802. 11

WLAN（无线局域网）的理念是将局域网的功能扩展到无线领域中，换句话说，就是用无线传输来取代电缆。如图 2-2 所示，移动计算机建立了到接入点（Access Point，AP）的无线通信，从而将设备连接到网络基础设施中。WLAN 提供了到互联网、公司网络、家庭网络的灵活便捷的接入，用户无需再使用以太网电缆，并且可以在一定的区域范围内自由地移动。

IEEE 的 802. 11 工作组定义了一系列无线局域网标准，对设备与 AP 之间以及设备和设备之间的物理层、媒体接入控制层（Media Access Control，MAC）进行了规范。第一个 IEEE 802. 11 标准发布于 1997 年，后来进行了一系列增强从而支持更高的传输速率和一些附加的功能。今天，IEEE 802. 11 已成为一个标准族（见表 2-2 和图 2-3）。

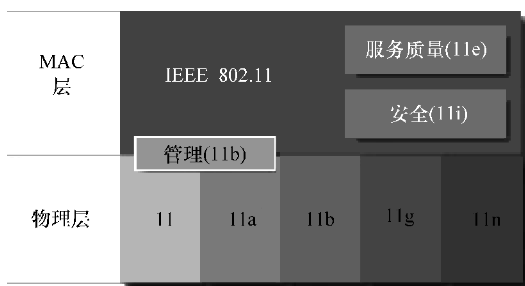


图 2-3　IEEE 802. 11 标准概览

1. 物理层

物理层定义了无线设备与接入点或其他无线设备之间的无线传输。最初的 IEEE 802. 11 标准在 2. 4G Hz 的工业、科学、医药（Industrial Scientific and Medical，ISM）用频段上支持高达 2Mbit/s 的传输速率，这个频段可以在世界范围内免费使用，无需牌照。同时定义了两种空中接口技术：直序扩频（Direct-Sequence Spread-Spectrum，DSSS）和跳频扩频（Frequency-Hopping Spread-Spectrum，FHSS）。在 DSSS 中，通过伪随机码片序列来实现数据的复用。由于码片的传输速率远高于数据的传输速率，调制后的信号的频带比原始数据要宽（所以称为扩频）。接收方使用相同的伪随机码片序列来进行解调。在 FHSS 中，也使用了类似的技术，但在通信过程中会快速地变换载频。

最初的 IEEE 802. 11 很快就被 IEEE 802. 11b 取代。IEEE 802. 11b 提供了更高的传输速率，可达 11Mbit/s，在室内的作用范围可达 30 ~ 50m，而在室外甚至可达 100m。它使用与 IEEE 802. 11 相同的频段，使用 DSSS 并能实现后向兼容。IEEE 802. 11b 使得 WLAN 技术在移动设备中被广泛采用。

通过 IEEE 802.11a 可以获得更高的速率，通过正交频分复用技术（Orthogonal Frequency Division Multiplexing，OFDM）可以达到 54Mbit/s 的传输速率。OFDM 技术将数据流分成很多不同的子流，并将不同的子流在不同的频率信道上同时传输。这种方式的主要优点之一是能够对抗选择性频率衰落。相比于 IEEE 802.11 和 IEEE 802.11b，IEEE 802.11a 工作在 5GHz 的频段上，并且不后向兼容 IEEE 802.11b，除非设备同时实现了两种标准，才能在两种模式下工作。更高的载频既有优点又有不足。一方面，5GHz 的频段相比与 2.4GHz 的频段使用得较少，从而少有干扰；另一方面，高频段的作用范围相对较小，且只能视距传输。

IEEE 802.11g 结合了 IEEE 802.11a 的高速特性，同时又能与 IEEE 802.11b 兼容。它工作在 2.4GHz 的频段上，且提供 54Mbit/s 的传输速率。该标准定义了 4 种物理层技术：DSSS（如 IEEE 802.11b）、OFDM、增强的 DSSS 和 OFDM 与 DSSS 的结合。

最新的 WLAN 标准为 IEEE 802.11n，它能达到 100Mbit/s 的传输速率并且有更远的传输距离，可通过多天线技术来实现这些特性。尽管 IEEE 802.11n 还在开发过程中，但目前已经有“准标准”的产品面世了。

2. MAC 层

MAC（Medium Access Control，媒体访问控制）层控制不同设备对共享无线信道的访问。IEEE 802.11 所定义的分布式协调功能（Distributed Coordination Function，DCF）能够使设备依照载波侦听多路访问/冲突避免（Carrier Sense Multiple Access/Collision Avoidance，CSMA/CA）协议对媒体进行竞争式的访问。

这种协议采用载波侦听的方式：设备需要传输数据时，首先对信道进行侦听，以确定信道是否有其他设备在传输数据。如果在一定的时间内，信道一直是空闲的，则设备可以启动数据传输。如果信道上有数据在传输，则设备将延迟发送。每个设备都有一个随机的计数器，当设备需要发送数据且信道空闲时，计数器开始倒数计数，直到计数器为 0 时，设备就可以发送数据。

尽管采用了这样的方法，仍然存在着多个设备同时发送数据的可能性，从而导致分组的冲突。为解决这个问题，则引入了可选的握手机制。其工作机理如下：如果设备需要发送数据，首先发送一个很短的“准备发送（Ready-To-Send，RTS）”的分组，包含了需要发送的分组的长度。若接收方当前没有接收其他数据，则发送一个“可以接收（Clear-To-Send，CTS）”的短分组。在这次握手完成后，发送方可以发送实际需要传输的数据。最后，接收方使用确认（ACK）分组来进行确认。

除了与接入点之间的通信外，IEEE 802.11 的 MAC 层还支持 Ad Hoc 的模式。设备之间可以直接通信而无需经过接入点中转。

3. 服务质量、安全与管理的扩展

IEEE 802.11 协议族还定义了服务质量（Quality of Service，QoS）机制。在 IEEE 802.11e 标准中，设备能够在 MAC 层为不同的分组分配不同的优先级，高优先级的分组能够比低优先级的分组更快地访问到媒体信道。其共定义了 4 个优先级：语音、视频、尽力而为和背景流量。语音具有最高的优先级而背景流量的优先级最低。

另一个扩展是针对 IEEE 802.11 在安全方面的扩展：IEEE 802.11i 标准包括了增强的加密和认证策略，如密钥管理和安全认证。最新的但并非最不重要的扩展是 IEEE 802.11h，在 IEEE 802.11a 的物理层和 MAC 层增加了针对频谱和功能控制的网络管理和控制扩展。

4. 总结

.11 系列（人们对 IEEE 802.11 的缩略称呼）是一个成功的实例。几乎每个新出的笔记本以及其他一些电子设备（如电视机和数码相机等）都装配了 IEEE 802.11 功能。在市场上一些多模卡产品能够支持不同类型的 IEEE 802.11 标准，有的产品还能够支持 IEEE 802.11 和 GSM/UMTS 的集成。

2.1.3 蜂窝网络：GSM 和 UMTS

蜂窝网络能够在广域上提供无线连接，如在一个国家的范围内，或甚至能够跨越国界。“蜂窝”指的是将区域划分成“小区”的一种划分方法。每个小区中都有一个基站提供无线服务。蜂窝是为了在空间上实现无线频谱的重用，每一个既定的无线信道能够在不同的小区中使用，同时发送而不受干扰，只要它们隔得足够远。蜂窝网络所使用的频率通常需要得到许可证才能使用。运营商获得一定的频段（一般通过政府指令）后，对网络进行维护和运营。用户签约后，可以使用移动电话来使用语音服务和数据服务。

1. 全球移动通信系统

第 1 代移动通信系统使用模拟信号传输，20 世纪 80 年代，在多个国家部署了一些不同的系统。第 2 代系统基于数字信号传输，开发和标准化从 20 世纪 80 年代后期开始，其中最典型的例子就是全球移动通信系统（Global System for Mobile Communication，GSM）。

GSM 的开发起始于 1987 年，欧洲的运营商和管理机构签署了一份会议记录，确认了引入通用的数字蜂窝电信系统的意愿。第一套 GSM 规范于 1990 年由欧洲电信标准化组织（European Telecommunication Standard Institute，ETSI）发布；第一个网络于一年后投入运营。在后续的几年中，GSM 在许多国家得以部署，其中也包括欧洲以外的国家，获得了巨大的成功。

GSM 所提供的主要服务是移动电话，质量堪比固定电话，并具备了一系列

的补充服务，如呼叫转移和语音邮箱等。除此之外，GSM 还提供低传输速率的数据服务，特别值得一提的是短消息服务（Short Message Service，SMS），用户能够通过短消息，以存储转发的方式实现文本信息的交换。从传输的角度来看，GSM 使用了频分多址（Frequency Division Multiple Access，FDMA）和时分多址（Time Division Multiple Access，TDMA）的接入方式，使用 900MHz 和 1800MHz 两个频段。在 900MHz 频段上，890 ~ 915MHz 用于从移动设备到基站的上行传输，935 ~960MHz 用于从基站到移动设备的下行传输，因此 GSM 是一个频分双工（Frequency Division Duplex，FDD）的系统，通过不同的频段来实现上行和下行的区分。每个频段又依频率划分为 124 个信道，每个信道 200kHz，一个信道分为 8 个时隙。每个移动设备可以分到一个给定的时隙，在这个时隙上能够实现 9.6kbit/s 的传输速率。

从网络的角度来看，无缝的连接和移动性可以通过基站之间的切换来实现，不同国家的网络运营商之间可以签订协议来实现漫游。为了实现这个目标，在网络中设置了一个专用的数据库，用于跟踪记录移动设备所在的小区。网络中的安全机制包括加解密、用户标识的保护和认证。安全方面的一个重要特征，是用户的认证数据（允许他/她对网络进行访问）不是存在移动设备上的，而是存放在一个小的芯片上，这个芯片叫做签约标识模块（Subscriber Identity Module，SIM），插入到移动设备中使用。用户可以使用不同的移动设备，只要 SIM 不变，都可以使用同样的用户号码。总之，GSM 标准定义了整套的系统和协议架构，包括针对语音和数据的物理传输、媒体访问、移动性管理、无线资源管理、服务、安全、互操作、网络管理以及其他一些功能。

自从 GSM 进入市场后，研发过程并没有停止。如语音编解码随后又得到了改进，从而提高了语音质量，群组呼叫和一键通（push-to-talk）服务被引入、数据传输技术也得以改进，包括更高的数据传输速率和基于分组的数据服务：

1）高速电路交换数据（High-Speed Circuit-Switched Data，HSCSD）服务通过同时使用一个频率信道上的多个时隙进行传输，使移动设备能够获得更好的数据传输速率（多时隙操作）。可以达到 10kbit/s 量级的传输速率。

2）通用分组无线服务（General Packet Radio Service，GPRS）在空中接口上引入了基于分组交换的传输机制，在不同的移动设备之间采用了一种动态的时隙分配机制，而不同于原来在一个会话或呼叫过程中，时隙仅为一个移动设备所使用的机制。GPRS 改进并简化了对互联网的访问，对网络的接入时间更短、数据传输速率更高、采用基于流量的计费且拥有“永远在线”的无线连接。

3）针对 GSM 演进的增强数据传输速率（Enhanced Data Rates for GSM Evolution，EDGE）相比于 GSM 采用了传输速率更高的调制模式，从而能够获得更高的数据传输速率（高于 100kbit/s）和更高的频谱效率。

2. 通用移动电信系统

并对第 3 代移动通信系统的标准化工作开始于 20 世纪 90 年代早期，当时将 2GHz 的频段分配用于未来的蜂窝通信。3G 致力于开发一系列的系统，被称为国际移动通信 2000（International Mobile Telecommunication 2000，IMT-2000），比第 2 代系统提供了更高的数据传输速率，并能提供互联网、多媒体服务和更加有效的频谱利用率。

来自欧洲、日本、中国、美国和韩国的不同公司、标准化组织共同组成了第 3 代合作伙伴计划（Third Generation Partnership Project，3GPP），并在后来开发出了 UMTS。UMTS 构建在 2G 的基础设施之上，但是引入了新的空中接口传输技术，采用了码分多址（Code Division Multiple Address，CDMA）的技术，定义了两种不同的模式：

1）采用频分双工方式的 WCDMA。

2）组合了 CDMA 和 TDMA 的时分双工模式。

第一个 3G 网络由日本的 NTT DoCoMo 于 2001 年 10 月启动，称为 FOMA（Freedom of Mobile Multimedia Access）。欧洲的运营商随后于 2002 跟进，UMTS 于 2005 年得到商用。

在 UMTS 中，空中接口传输速率理论上峰值已经达到 2Mbit/s。实际上，只可以达到数百 kbit/s。作为一个新的空中接口，UMTS 不能与 GSM 兼容，UMTS 的终端通常都是 UMTS 和 GSM 双模的，包含了从 UMTS 到 GSM 的自动切换机制（反之亦然），当一个网络的覆盖衰落时可以切换到另一个网络。

得益于数据传输速率上的提升，原来在 GSM 中无法实现的一些数据服务现在可以在 UMTS 中使用了。UMTS 用户可以浏览网页，使用多媒体消息服务发送和接收包含多媒体信息的消息，此外，还提供了基于流媒体的电视服务、视频电话和音乐、游戏的下载等。实际上，移动电话已经成为一个多用途的多媒体设备，它集成了电话、音乐播放器、照相机、收音机、视频游戏、小电视、闹铃等多种功能。

向更高传输速率的演进还在继续，如高速分组数据接入（High Speed Packet Data Access，HSPDA），这是 WCDMA 中一种新的基于分组的下行技术，有望能够提供 10Mbit/s 的传输速率。HSPDA 提供了自适应的调制和编码技术，采用了多天线技术。

3. 服务平台

蜂窝网络上的一个重要增强是引入了服务平台。在服务平台上并不对服务进行标准化，而是对生成服务的服务特征进行定义。通过服务平台，运营商可以方便、快捷地引入新的服务，这有利于在标准服务的基础上进一步提供差异化的、运营商特有的增值服务。GSM/UMTS 定义了以下服务平台（更多细节可以参见

第3章)：

1）移动网络增强逻辑的定制应用（Customized Application for Mobile Network Enhanced Logic，CAMEL）将智能网（Intelligent Network，IN）的功能引入到了GSM中。借助CAMEL，运营商能够提供更多的补充服务，如800电话、预付费电话和短号服务等。

2）SIM应用套件（SIM Application Toolkit，SAT）是一种在SIM卡中执行应用的机制。基于SAT，可以在移动终端上显示运营商定制的菜单、图标并播放音频，如用户可以下载新的铃声到移动电话中。

3）无线应用协议（Wireless Application Protocol，WAP）为移动用户引入了一个类似于Web的信息平台，为传输和显示用于移动用户的类似于Web的服务定义了一套系统架构、协议族和应用环境。签约用户可以下载如新闻、天气预报、股票信息、当地城市信息等内容，还可以提供电子商务服务如票务预定和电子银行等。另一种类似于WAP的服务平台是i-mode，由NTT DoCoMo开发并首先在日本的蜂窝网络中提供服务，随后输出到其他国家并在GSM网络中也得以部署。不同于WAP，i-mode从一开始就非常成功，其成功源于引入了内容提供商角色的创新的商业模式。

4）IP多媒体子系统（IP Multimedia Subsystem，IMS）是UMTS的一部分，用于控制基于IP的多媒体服务，如VoIP。IMS在基于分组交换的网络中组合了语音与数据。

2.2 无线网络一览

这个部分对基于蜂窝的无线网络进行介绍，目的是使读者从用户或者应用开发者的角度来形成对移动网络的基本了解。我们对主要的架构和功能进行了介绍，特别介绍了移动性管理、寻呼和切换这几个问题，但并不打算介绍整个协议族。因此将不会详细介绍无线传输技术并省略了很多与无线相关的重要的控制功能，如功率控制、无线资源管理、天线技术等。

我们可以看到，未来的无线架构应该可以兼容多种无线技术，包括WLAN热点等。通过IP将这些不同的技术、异构的网络连接起来。

泛在计算设备/网络也是未来的一个发展方向。目前，移动电话、PDA、便携PC等都已经连入到无线网络。在未来，更多的嵌入式设备将具备无线通信功能，如家庭中的火警检测设备、娱乐设备、公共场所的传感器等。火车中的温度传感器就是一个例子。另一种重要的无线设备是RFID标签，一旦被触发时可以传送一些标识信息，使得各种设备能够在当前的环境中访问到一些信息。

由此，无线网络将成为日益扩展的数字世界的基础。著名的Metcalfe定律认

为，通信网络的价值与网络规模的二次方成正比，Sarnoff 定律认为广播网络的价值正比于用户的数量。但在泛在网络中，由于大部分设备都是被动的，网络的价值与网络规模的关系还有待确定。信息的价值取决于它们的存取范围和时限，这些因素对于泛在设备来说受到一定程度的限制。如果增长率高于线性，但低于二次曲线，大量的设备将会使网络的价值获得巨大的增长。

2.2.1　蜂窝网络的架构

如 2.1 节所提到的，无线网络通常是基于蜂窝结构的。本节将讨论这类网络的系统架构，包括了 2G 网络中的 GSM 和 3G 网络中的 UMTS。蜂窝网络的简图如图 2-4 所示，由接入网络和核心网络构成。从组网的观点来看，两个部分都提供了无缝的服务连续性。在不连续的无线链路上，或者需要在基站之间进行切换的情况下提供了不会被移动性中断的服务。

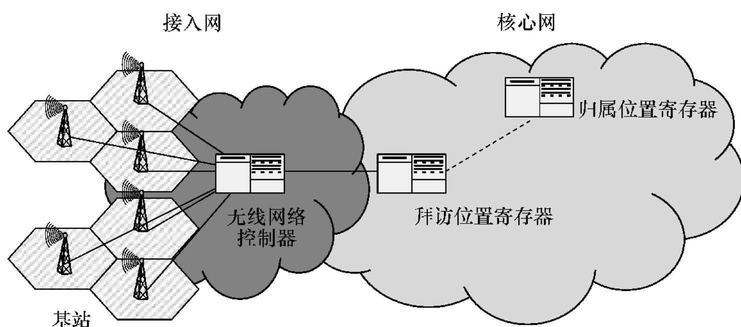


图 2-4　第 2 代蜂窝网络

接入网络具有以下功能：

1）移动设备向网络附着。

2）本地的移动性管理，如在本地范围内寻呼位置未知的移动设备。

3）在基站之间切换正在进行的通话或会话。

4）无线资源控制，如接入网中物理层资源的管理。

5）链路层安全，包括用户的识别和访问控制等。

如图 2-4 所示，接入网的功能在基站控制器或无线网络控制器（Radio Network Controller，RNC）中实现。

核心网有如下功能：

1）网络与网络之间的全局移动性，包括通过与位置无关的编号（例如电话号码）来实现寻址等。

2）用户证书的管理以及网络准入控制。

3）在不同的网络之间对服务的漫游的支持。

4）对数据分组和呼叫处理信令进行路由，即对控制面和用户面的流量进行路由，包括名字的解析等。

5）为第三方提供服务控制接口，如由外部始发的呼叫建立。

为了更好地理解这一领域的专业术语，可以参考 RFC 3753。

2.2.2 移动性管理和切换

移动性管理指的是处理用户移动性的网络能力。移动性又区分为无缝移动性和游牧移动性。前者在用户移动时提供服务连续性，而后者指移动节点能够在不同的地点附着到网络中，但并不需要提供持续的（用户节点的）可达性和服务的连续性。

移动性管理中的主要技术问题如下：

1）移动设备在不同地点的可达性。

2）服务进行中时，在基站之间的无缝切换。

在移动网络中，为了保证可达性，使用了两个地址：一个固定的与位置无关的地址和一个用于查找服务的与当前的位置相关的临时地址。在 GSM 中，归属位置寄存器（Home Location Register，HLR）和拜访位置寄存器（Visited Location Register，VLR）保存了移动设备的当前位置，从而使呼叫可以被路由到这个位置上（见图 2-4）。

无缝切换意味着，当一个用户从一个基站的覆盖区移动到另一个基站的覆盖区时，不会发生明显的服务上的中断。为了减少切换所导致的服务中断的时间，已经进行了大量的研究，特别是针对语音呼叫等交互式服务，这类服务对于数据的丢失最为敏感。

可以通过切换所使用的技术和切换中管理域边界的划分对切换进行分类。如果切换前和切换后的网络使用了不同的无线技术或不同的组网方式，则称为不同技术之间的切换。另一种类型是在不同网络域之间的域间切换。对于域间切换，如果两个域之间没有交换用户证书等用于认证或传递信任关系的数据，则需要在切换时对用户进行重新鉴权。在这两种情况下，无缝切换都是很难实现的。

根据切换发生时所体现的不同的网络能力，可以将切换分为以下几种类型：

1）硬切换，也称为先断后建的切换，在连接到新的基站之前首先释放与老的基站之间的连接。这是一种最简单的切换形式，可能会导致服务的延迟或数据的丢失。这也是一种最基本的形式，在无线通信中，与老的基站连接中断通常不会得到提示。

2）先建后断的切换，终端与切换前的基站和切换后的基站同时保持连接。通过这种方式，可以保证数据能同时在不同的数据通路上传递，分组数据也不会丢失。然而这种方式所面临的一个比较实际的问题是，在很多移动系统中，终端

由于基站使用不同频率以及其他一些原因，不能够同时连接到两个基站上。

3）预先切换是一种“有准备”的硬切换。在与切换前的基站断开连接时，已经将到切换后基站的连接准备妥当。如可以先在切换后的基站上注册，从而提前完成证书的交换等动作。

预先切换能够避免寻找合适的目标基站并在新基站和底层网络分配资源的问题。这对于不同技术之间的切换以及跨域的切换尤其有用。在前面讲到的两种场景下，预先切换需要获取新网络的信息，从而可以找到最合适的网络。有些方案通过原网络来与新网络取得联系。

有很多方式可以用来支持先建后断的切换。在最简单的情况下，移动设备同时连接到两个网络。设备有可能同时连接到两个使用不同技术的网络，从而需要两套收发装置，这称为网络多样性，需要支持将正在进行的会话有效地从一个网络转移到另一个网络的能力。另一种情况被称为宏多样性，从物理层支持上面所提到的切换。如在 CDMA 系统中，设备可以同时和两个基站通信，这种类型的切换也称为软切换。

2.2.3　寻呼

另一个与移动性有关的概念是寻呼。为了对呼叫或数据会话进行路由，网络需要首先对移动设备进行定位（小区）。为了能够在后面被定位到，移动设备首先需要向网络发送位置更新消息，由于移动设备的电源电力有限，位置消息发送应该考虑到效率和能量消耗等因素。为了减少信令的开销和不必要的能量消耗，可以将几个小区组成一个寻呼区，处于空闲状态的移动设备在切换小区时不需要向网络发送位置更新消息，只有在移动到一个新的寻呼区时才需要向网络发送位置更新消息。当一个呼叫到达网络时，网络向设备所在的寻呼区的所有小区发起寻呼，小区中所有的基站都将从一个独立的信令信道向移动设备进行查询以确定设备的位置。

在寻呼中需要用到几个概念，如地毯式问询和顺序寻呼。在地毯式问询中，寻呼请求同时发送到寻呼区中的各个小区；而在顺序寻呼中，寻呼请求按照一定的次序发送到各个小区，如按设备在某个小区中的概率排序。所有的策略主要考虑如何折中位置更新的效率和联系到目标的准确率。

2.2.4　漫游

运营商之间的漫游协议能够确保用户在这些运营商的网络中使用服务，从而能够在一定程度上保证用户的移动性，但这种漫游协议通常是用于游牧的场景的。从技术上来说，就是要求拜访网络的运营商能够在无法完全获取用户认证信息的情况下也能够准入漫游到本网络的用户。之所以会有这样一种场景是因为运

营商之间通常不能交换用户的共享密钥或者根密钥等信息，因此拜访网络的运营商只能将认证、准入控制这样的任务委托给用户的归属网络，或通过归属网络所提供的有限的信息来进行检查。一种常见的方法是向归属网络请求一个质询-应答对。拜访网络可以将质询发送给用户，用户通过设备内置的密钥对该质询形成一个应答，拜访网络将用户反馈的应答与归属网络发过来的应答进行对比，从而实现用户身份的检查。在这种方式中，只需要进行一次与归属网络的信息交换。

除了认证之外，网络还需要对用户进行服务使用方面的授权并谈妥运营商之间的结算汇率。这些内容通过预先设定被称为服务等级规范（Service Level Specifications，SLS）的双边协议来实现。

2.2.5 服务质量

服务质量（QoS），是一个在网络和无线系统中广泛使用的术语。这中间的服务指的是终端用户的服务，而 QoS 则用在不同的层面上。

QoS 是一个分层的系统模型，在大多数的通信系统中，需要谈及系统的特定层面。在应用层中，指用户所感受到的真实的质量，很明显这是和应用相关的，是用户所感受到的质量的测量。对于应用而言，很难将所测量的质量转换成技术参数，如连续的数据丢失通常比零星的语音编码的丢失要严重得多。另外，这些测量基于很多其他因素，如通信所使用的语言。有些应用层的参数则容易测量一些，如呼叫建立时延或掉话率等。

对于网络层，QoS 通常指用户服务的端到端可用的传输能力，包括数据传输速率、时延和数据丢失率。由于应用的需求会发生动态的变化，网络也会承载一些服务的变种，因此会使用一些统计上的数据，如最大的偏移量、平均丢失率、最大的数据传输速率等。为了使网络上的服务具有一些结构性，可以根据应用的需求将它们分成一些类别。一个典型的分类建议是 ITU-T 的 Y. 1541，其分类如图 2-5 所示。在链路层和物理层，QoS 以误码率或点到点的链路容量来测量。

	0级	1级	2级	3级	4级	5级
时延	100ms	400ms	100ms	400ms	1s	未规定
时延抖动	50ms	50ms	未规定	未规定	未规定	未规定
丢包率	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-3}	未规定
误码率	10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-4}	未规定
业务实例	语音，视频	语音，视频	信令	交叉数据，信令	视频流，大块数据	尽力而为数据

图 2-5 Y. 1541 定义的 QoS 分级和门限值

当前主要的蜂窝系统都是针对语音进行优化的，这意味着不同层上的延时和差错处理都是针对语音的。业界为了给数据服务提供恰当的 QoS，也投入了相当的精力。一个主要问题是，预先并不知道所需的数据传输速率而呼叫或会话建立通常会需求较高。应特别指出，像寻呼这样的技术适合于语音呼叫但可能会产生很高的会话延时，对于数据应用是不合适的。另一方面，一旦会话建立起来，很多数据应用就对数据丢失和延时不再敏感了。

2.2.6　位置服务

虽然移动通信网络本质上来说是提供通信服务的，但也可以提供其他服务如位置服务。通过基站信号的三角测量，网络可以对用户进行定位。位置信息的准确性取决于网络的覆盖情况，运营商可以自己使用这些位置信息，也可以提供给第三方的服务提供商使用。

另一个位置系统是协作型全球定位系统（Assisted Global Positioning System，A-GPS）。该系统是 GPS 在手机使用中的一个变种。设备从基站中接收 GPS 卫星信息（包括轨道、频率和功能），从而设备的 GPS 接收器能够检测到非常微弱的卫星信号并对其进行分析。相比于传统的 GPS，A-GPS 处理速度更快，且更节省电力。

通过这两种技术，运营商和第三方的服务提供商都能够为用户提供位置服务（如导航服务、基于位置的紧急呼叫、推荐附近的餐馆等）。

2.2.7　广播和组播服务

另一种很有意思的服务是一点对多点的通信服务，如内容的广播或组播服务。出于节省资源的目的，将数据同时发送到多个目的地有利于节省无线链路资源的使用。对于发送者来说，由于减少了会话的个数，也带来了一定的好处，如提高视频服务器的扩展性。一点到多点的内容分发是基于用户在同一时刻所要的内容是相同的这样一个假设。

组播可以分成两个功能的组合：组播数据的分发和群组的管理。两个这方面的重要协议是互联网多播协议[2]和 3GPP 多媒体广播和组播服务——MBMS[3]。后者在无线网络的环境中对前者进行了优化并集成到 UMTS 的系统架构中。

在移动网络中的多播服务需要考虑以下几个因素：第一，分组管理必须考虑到用户的移动性；第二，媒体的分发需要考虑到所使用的无线技术。这件事情可能会比较复杂，如不同用户终端上的功率控制都是特定的，对于一组节点不能同等对待。

2.3 基于IP的下一代移动网络

固定与移动网络演进的趋势是互联网技术逐渐占据主导地位。在本节中，我们将讨论基于IP的下一代移动网络的理念和技术。

在2.1节中我们曾经提及，一些不同的无线技术在覆盖范围、数据传输速率等方面形成互补之势。下一代移动网络的一个主要问题就是要考虑在异构环境中无缝的无线接入。IP将作为在不同无线技术之间交换数据的“中间语言”，这些无线技术包括WLAN、3G和即将到来的4G。为了支持异构接入，移动网络在向全IP的方向发展，以获得可扩展性和成本上的节约。基于IP的下一代移动网络的一个示意图如图2-6所示。

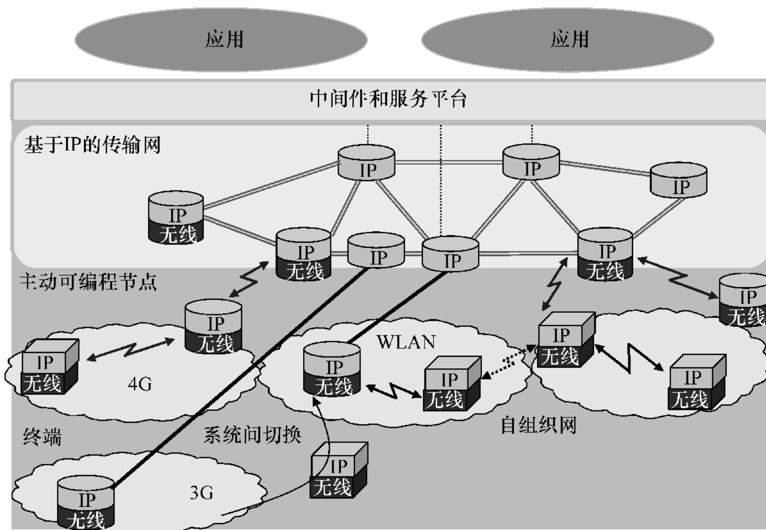


图2-6 基于IP的移动网络

IP协议族的成功在于它遵循了一些基本原则。互联网仅提供了基本的连接作为它向上层提供的服务，并且不提供服务保证。对于万维网以及其他数据应用来说，这点是简单且有效的。互联网的架构基于端到端的原则，尽可能地将控制转移到端系统之中。这条原则减少了网络中的状态，减少了网络的成本，提高了在终端中安装功能和应用的灵活性。当固定网络中的带宽变得日渐便宜，使用起来不再昂贵到让人却步，互联网的架构便以其简单性而获得了成功。这里的互联网架构指的是RFC3724[4]中所描述的架构。互联网架构的设计及其在社交、经济以及政治领域所带来的冲击在Clark等论文[6]中有所讨论。

有几个原因使得移动运营商将网络向基于IP的架构迁移。其一，IP是基于

分组的传输机制，能够灵活地用在不同应用中；其二，IP 是一种广泛使用的组网技术，独立于各种特定的无线技术。

另一方面，现在的互联网将移动网络以及移动性等问题放在较低的优先级上，这是未来移动互联网将要重视的一个问题。由于移动互联网需要处理用户的移动性以及有限的无线资源，因此需要增加一些新的功能。

2.3.1　异构接入和移动性

不同的运营商提供了多种基于不同技术的无线网络，用户可以自由地选择技术及其所提供的服务，因此无线通信的使用变得更加廉价。另外，新网络的投资成本也在下降。另一方面，网络也需要集成不同技术所提供的能力，从而能够为用户提供端到端的、无缝的、安全的解决方案。

异构的无线系统将对整个网络都有很多影响，包括对服务和媒体的分发。最大的挑战来自于要在不同的无线接入技术上实现无缝的、优化的服务。为了在异构环境中提供面向用户的服务，网络应能具备一定的智能，代替用户来进行网络的优化。如通过使用关于用户、网络和服务的上下文信息来提供最好的网络服务。通过这样的方式，网络可以提供无缝的、面向用户的服务，用户并不需要感知到不同的无线接入网络。

如有时为了提供更好的服务，需要切换到一个能够提供更多网络能力的接入点上去。当经过一个无线环境热点时，在短时间内，可以切换到这个接入点上获取所需的服务，如下载大块的数据或进行视频会议。然而在大多数情况下在执行切换前，切换目标点的资源情况是不可预知的。

基于 IP 的移动性管理

最终，互联网的设计仅针对永久性地连接在上面的节点，用户具有固定的 IP 地址，连接到固定的端口上。在这种情况下，IP 地址既用于标识网络上的节点又用于到这个节点的路由。移动性提出了对用于数据路由的定位符和标识符进行分离的需求，从而引入了几种对互联网协议进行扩展的方法。

最主要的方式是移动 IP[7,8]，它使用两个 IP 地址，一个归属地址用于标识设备而另一个转交地址用于描述设备当前的位置。发送到移动设备的分组将首先发送到归属地址上，再由归属代理将分组转发到转交地址上。

移动 IP 的基本原理如图 2-7 所示。最主要的网络实体是归属代理，驻留在移动设备的归属网络中。移动设备需要在归属网络中注册转交地址，归属代理将拦截发送到该移动设备的数据并转发到移动设备的转交地址上。为了实现这种中转，需要使用隧道来转发数据。

移动 IP 的优点是可以从现有的网络迁移过去，因为只增加了一个新的实体，无需修改 TCP/IP 或者其他设备上的协议栈。并且，因为使用了隧道，数据本身

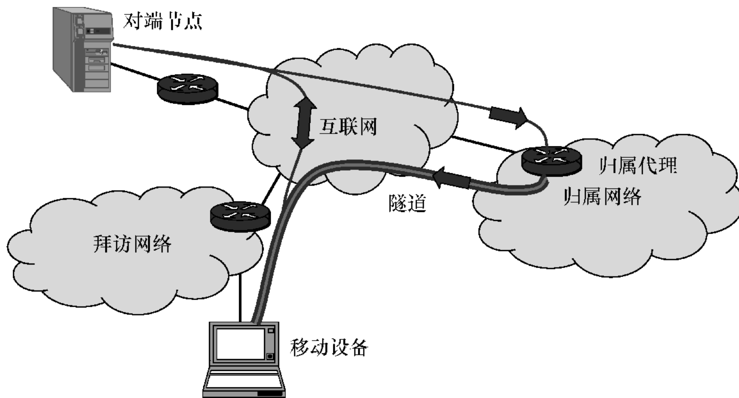


图2-7 移动IPv6

也不会受到影响。由于IP是独立于无线技术的，因此移动IP提供了一种可用于异构网络的解决方案。

另一方面，移动IP也有不足之处，存在着改进或使用其他替代方案的空间。首先，数据通路不是优化的，尽管IPv6的路由优化使用了一些更加直连的路由，但这需要与之通信的节点感知到移动设备的转交地址并且跟踪该设备的移动，这就意味着通信节点将获知移动节点的位置和移动情况，从而带来隐私方面的问题。

移动IP的另一个不足是不支持有效的切换，因为所有的位置更新都需要回到归属代理上。这点很像前面讨论过的移动核心网，它们的目标都在于可达性。由于不能对蜂窝环境中频繁的本地切换进行有效的支持，IP移动性解决方案分成了两个部分：

1）宏移动协议，如移动IP，在一个很大的范畴内，针对连接到互联网的不同无线网络实现的IP的可达性。

2）微移动协议则用于管理一个拜访域内部的移动性，并对频繁的切换进行了优化。目前一些微移动协议正在开发当中[9]。这些方案的优势在于，域内的切换对外部是不可见的，从而降低了切换延迟和信令负载。但另一方面，这些方法产生了一些移动特性的状态信息，需要在节点和网络之间使用额外的安全关联。

对于像互联网这样的开放网络，移动性会带来一些新的安全威胁，如一个设备可以在切换期间或者路由优化的过程中劫持会话。在这些情况下，一个恶意的设备可以监听或截获流量，从而可以伪装成另一个移动设备并将本不属于自己的流量导向自己[8]。目前来看，主要的挑战是要解决这些问题但又不能依赖于蜂窝网络的安全基础设施。移动IP所存在的这些问题目前正在逐步得到解决。

另一种 IP 移动性管理的方法是 IP[2] 架构[9]，在这种架构中节点不需要参与到移动性管理而由网络全权代劳。与移动 IP 相比，IP[2] 架构定义了一个新的移动性管理层，进行路由管理和新加入的位置管理功能，能对移动 IP 进行一些改进，包括路由优化和位置隐私的保护。另一方面，这种架构秉承了互联网面向端设备的理念，基于端到端的原则来进行系统设计。

最新的成果采用了一种在 IP 层和应用层之间加入一个命名层次的方法，为此 IETF 定义了主机标识协议（Host Identity Protocol，HIP）。[11] 在这种机制中，在一个全新的命名层中使用了加密的标识符。两个设备需要通信时首先在安全标识之间建立安全关联。这种机制可以用于避免 DoS 攻击，并且可以用于移动性控制。这种方法也源自互联网的设计理念，数据传输不依赖于会话的建立。

2.3.2　IP 服务质量

互联网的服务质量作为一个课题已经被研究了 15 年，一些解决方案被提出并标准化。随后，我们将介绍一些主要的 QoS 方法，在介绍这些方法时，我们将对数据平面和控制平面加以区分。

数据平面也称为用户平面，QoS 需要确保某个服务等级或数据流的数据包获得约定好的、应确保的服务或从统计上应达到服务质量的要求。控制平面的职责是对数据通路进行配置并为用户的 QoS 服务请求提供信令支持并在请求被许可的情况下提供网络能力支持。

在数据平面，集成服务（Integrated Service，IntServ）QoS 机制对端到端的每个流进行资源预留。一个流通常是由应用产生的一个数据流，可以通过源端和目的端地址以及端口号标识，端口号用于区分应用。对于服务的保证，集成服务规范了不同的保证级别。对于负载的控制，采用较为软性的基于统计的保证。为了保证端到端的资源预留，数据通路上所有的路由器都需要参与。这种资源预留的操作通过控制平面来实现，使用 RSVP。[12]

另一种主流的 QoS 机制是 IETF 定义的区分服务（Differentiated Service，DiffServ）。不同于在网络中的每个路由器都保留流的状态，区分服务将 QoS 保证分配给流量的聚类，这种聚类在网络边缘生成，通过在 IP 头的特定字段——DS 字段进行标记来实现。这种标记称为区分服务点码，提供了针对数据包的 QoS 需求的信息，需要使用几个比特。DiffServ 规定了一些被称为一跳行为（Per-Hop Behaviours，PHB）的规范。这意味着，针对一个分组只能逐跳地进行处理，而不是端到端的。因此，区分服务模型并不能用于端到端的情况，但端到端的服务必须构建在区分服务模型之上。对于聚类的处理，而不是处理单个流使得区分服务具有更好的伸缩性，而付出的代价是不能对每个流进行精细的管理。

IETF 所定义的两种主要的一跳行为是预期转发（Expedited Forwarding，EF）

PHB[13]和担保转发（Assured Forwarding，AF）PHB。EF PHB 实现了一种类似于租用专线的服务，带宽不能超过给定峰值。真正的 EF PHB 服务实现在网络边缘需要一些额外的真对入口流量的控制，在租用专线服务的数据路径中需要保存网络状态。IETF 没有对这些实现进行标准化，但可以通过使用带宽代理或如下面讨论到的一些其他信令来实现。

AF PHB[14]不提供带宽保证，但数据包在每一跳上都被赋予不同的优先级。在网络拥塞的情况下，使用担保转发的用户将会更少碰到尽力而为用户所面临的带宽减少的情况。

担保转发有4种服务类别，由路由器的不同队列来实现。每一个类别又有3种级别的丢弃过程：低、中、高。这些丢弃级别与服务类别是垂直的，可以在一个队列中对数据包进行区分处理。在同一服务类别中需要按照不同优先级进行丢弃处理的数据包不需要重新排序。很多服务需要进行排序处理，如多媒体内容服务，数据之间的重要性不尽相同，就需要进行排序处理。在同一队列中对数据包进行差异处理能够减少排序的过程。

另一种 QoS 机制是多协议标签交换（Multi-Protocol Label Switching，MPLS）。这是一个2.5层协议，每个流用流标签来进行标识，流标签插在第二层和第三层头之间。MPLS 提供基于流的路由，既能用于流量工程又能用于 QoS。对于流量工程来说，流能够独立于 IP 路由进行转发，使得网络设计者能够在网络中控制真实的数据流。对于 QoS，可以通过扩展来对每个 MPLS 流指定 QoS 参数，并使用简单的硬件来支持。MPLS 的概念也是从 ATM 虚电路发源而来，与其相似。

对于控制平面的 QoS，有两种主要的类别：逐跳处理和 off-path 信令。

逐跳预留方法基于伴随着数据流的数据通路的端到端信令，这种信令方式也称为 on-path 信令。逐跳预留的架构特性如下：

1）QoS 资源在每个路由器本地进行管理。

2）信令由终端触发并一路伴随数据通路。

3）端到端的资源预留通过在每一个路由器保留状态的机制逐跳建立。

普遍认为，逐跳模型的弊端在于伸缩性，这是因为每个路由器都需要处理每个流的状态信息。这里有三种代表性的架构，即 RSVP、MPLS 信令和 NSIS。RSVP 是 IntServ 的控制协议，基于 IntServ 的流的概念。它是一个第三层的协议，并在数据通路的每个路由器上建立状态信息。对于组播的扩展和数据通路的重路由具有一定的灵活性。最近，IETF 的“nsis”工作组[15]开始启动 QoS 标准化工作。其目标是定义更为通用的 QoS 信令协议，可以支持不同的 QoS 技术和其他信令功能。

Off-path QoS 控制架构基于网络中称为资源管理器（Resource Manager，RM）的专用实体，该实体也被称为带宽代理[16,17]，其特征如下：

1）通过一个单独的资源管理器来处理每个域中的资源。

2）资源管理器维护着每个域的最新的资源情况和资源预留情况。

3）资源通过资源管理器进行预留，并由资源管理器来执行准入控制。

这种概念的优势在于，不需要太多的网元介入到 QoS 请求的处理中。特别是对于不需要严格保证 QoS 的服务，无需在每个路由器中建立状态，由 RM 对整个网络的拥塞状态进行监控并基于它的准入控制来进行处理。RM 也可以基于 EF DiffServ 模型来实现有严格保证的 QoS 服务。这是流量在网络边缘设备即网络的入口点上进行控制，并被分配以 PHB。对于网络的每一个连接，都需要对 EF 进行资源预留，RM 对它所许可给流使用的资源使用情况进行检测。

off-path 信令方案的潜在缺陷在于信令与数据通路不是耦合的。信令的故障或者数据通路的故障是不相关的，使得差错处理更为困难。

尽管各种各样的方案被提出，但目前还没有一种方案被广泛地部署到固定互联网中，为端到端服务提供保障。因为固定网络中的带宽比较便宜，目前主流仍然在使用尽力而为和恰当的带宽提供相结合的方式。对于 Web 浏览和诸如语音呼叫之类的低带宽应用来说是够用的。很多人对为什么 IP QoS 没有被真正采用进行了讨论[18,19]，认为主要原因是商业模式的问题和市场需求的问题。随着应用需求的增加和移动网络的发展，我们预计对于 QoS 的要求将会增长，其收益模型也会成为现实。

移动互联网的 QoS

基于移动 IP 的网络中，QoS 信令的主要需求如下：

1）与不同的用于无缝切换的移动性管理机制互操作，包括跨域的切换。

2）独立于在数据通路上提供 QoS 的特定 QoS 技术。

3）独立于特定的无线接入技术。

使用互联网的协议意味着能够满足其中的一些需求，但有一个特殊的问题是移动性和 QoS 的集成。关于这方面的一些方法的描述可以参考 Manner 等的文章[20]，其中对资源和移动性管理之间的交互进行了一些分类。

QoS 信令需要进行一些扩展来支持不同的切换模型。例如，对于 on-path 信令，在切换时旧的和新的数据通路上的路由器需要相互协调，因为它们都包含了数据流的状态信息。对于预先切换而言，从预先切换的接入点到相应的节点之间的资源需要进行预留。图 2-8 展示了预先切换的过程[21]，图 2-9 展示了网络中各个域通过 RM 使用 off-path 信令的情况。在这个例子中，经过基站（1）的信令可以通知新网络（2）为即将发生的切换（3）进行资源的准备工作。切换成功后，旧的资源被释放。通过这种方式在接入到新的网络之前实现了资源的预留，资源预留的触发既可以通过终端节点又可以通过网络智能（如对于移动的预测）。

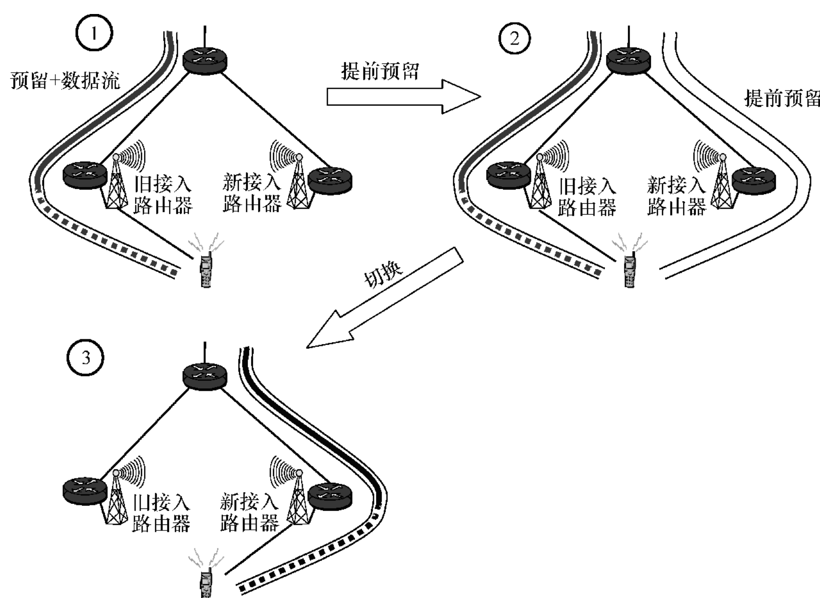


图 2-8 QoS 预留和预先切换

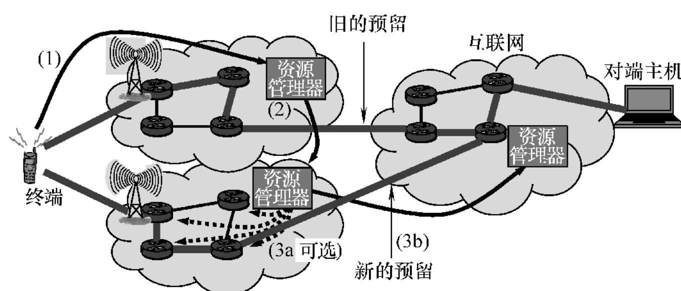


图 2-9 基于域的 QoS 控制和预先切换

2.4 泛在计算和自组织组网

在过去的几年中，电话和笔记本电脑的无线连接变得非常重要。典型的例子是 GSM 的巨大成功和 WLAN 的广泛部署。现在，越来越多的便携设备加入到网络中来，泛在计算成为一个趋势。泛在计算意味着计算机被植入到日常用品当中，如电子书、身份证、冰箱、汽车和洗衣机等，使它们具备了先进的功能。泛在计算的愿景是计算机融入到我们周围的环境之中，所有的设备都通过无线技术连接起来。这种愿景很早之前就由 Weiser[22] 提出，他认为计算机将被普遍地植

入到各个地方，传统的计算机将消失。

泛在计算不仅创造了新的应用，还需要使用新的无线通信技术和协议。这里的关注点并不一定是更高的传输速率，能量的节省、小型化、分布式网络的组织形式、系统的规模、设备的低成本等将是更重要的因素。相比于今天的无线网络，泛在计算所需的网络将更加具有动态性和异构性的特点。在 GSM 和 UMTS 中，移动设备的通信都要经过基站，网络基础设施是集中管理的，不适合于泛在计算。我们需要能够使设备之间直接以对等方式进行通信的无线技术来支持泛在计算。设备之间应能够自发地组成网络而无需建立和维护网络基础设施，这被称为自组织组网（Ad Hoc Networking）的无线通信技术能够支持这种模式。在自组织网络中，无线设备建立起一个无需基站或其他预先部署的网络基础设施的网络，而是设备自己相互之间组织成一个网络。一个明显的特征是无限的多跳通信：如果两个设备不能建立起直接的无线连接（由于它们之间相互距离太远），它们之间的设备可以作为中继，在源和目标之间中转数据。如果在无线多跳组网中，大部分设备是固定的，通常也称为网状网（Mesh Networking）。

接下来的部分将描述一些泛在计算、自组织和网状网的应用场景，覆盖了移动计算机之间的自发（自组织）网络、无线传感器、可穿戴计算和汽车之间的网络。

2.4.1　移动自组织计算

可能自组织组网最直接的应用就是移动计算机之间的无线连接，如在会议中共同操作一个文档或访问投影仪、传感器之类的共享设备（见图 2-10）。在这些无线连接上引入多跳能力能够使网络容纳更多的计算机，作用于更大的范围。如

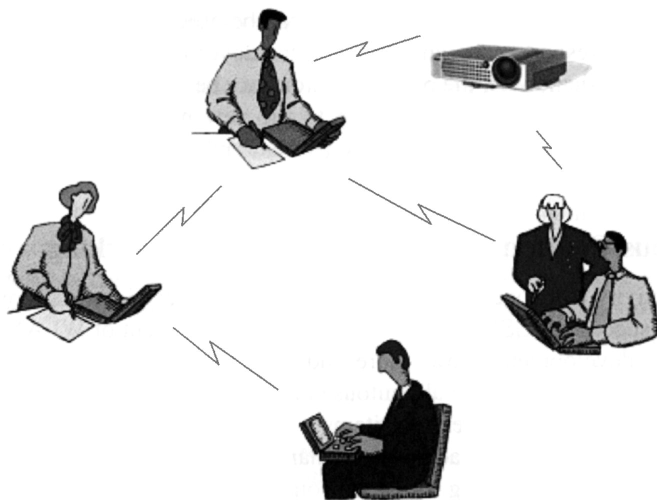


图 2-10　自组织移动计算场景

在校园范畴内的自组织网中，消息能够在计算机和计算机之间以逐跳的方式传递。在网络基础设施不可用、太昂贵、效率差或被毁损时，也可以找到应用场景。例如当自然灾难毁坏了技术基础设施的情况下，人道援助组织需要鲁棒的网络来协调医药救援和现场清理，自组织网络就能提供这样的帮助。

一些开发和标准化工作正在将这些场景变成现实。IETF 已经开发了一组基于 IP 的移动自组织网络路由协议，包括 Ad Hoc On-Demand Distance Vector（AODV）、Dynamic Source Routing（DSR）、Optimized Link State Routing（OLSR）和 Topology Dissemination Based on Reverse-Path Forwarding（TBRPF）。尽管底层的路由机制与 IP 路由相似，但自组织协议能够迅速对故障和设备的移动性做出反应。与蜂窝网络中的移动性相比，它们通常在完全非中心化的情况下进行操作，即没有一个集中的位置寄存器或其他集中的数据库。由于用于路由决策的转发表的信息都从网络拓扑信息导出，这些协议通常被归为基于拓扑的路由。另外一种有意思的路由方式是基于地理的路由，这种方法还没有进入到标准化的进程，与基于拓扑的路由不同的是，这种路由决策是基于邻居节点的位置和目的节点位置的信息。数据包转发给邻居，然后由邻居转发给目的节点（见图 2-11）。这种方法需要节点感知到它们的位置。与基于拓扑的路由相比，这种方法的优势在于，不需要在网络节点中维护路由信息，在高度移动的网络中体现更为明显。这种协议特别适合于位置信息一直可用的网络（如2.4.5 节所讨论的汽车到汽车的通信）。地理路由也可以用于2.4.2 节将要讨论的网状网。然而，地理路由只有在节点间的物理距离能够很好地作为连接的度量的情况下才能较好地工作。在更为复杂的无线传播环境中，基于反应连接情况的虚拟坐标系统的路由会比基于实际地理位置的路由工作得更好[23]。

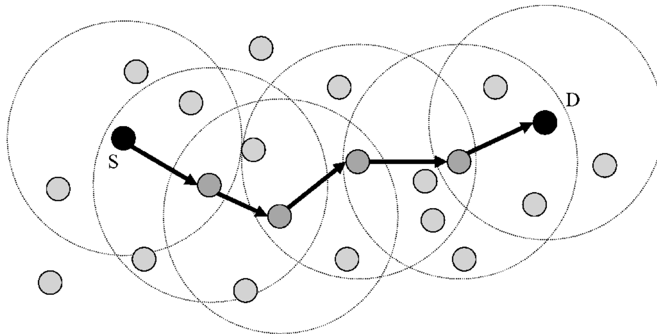


图 2-11　从源节点 S 到目的节点 D 的地理路由

上面所讨论的工作多基于 IEEE 802.11 作为无线媒介的情况，此外蓝牙也是小规模自组织移动计算的可行技术。

2.4.2　多跳无线接入网和网状网

通常认为无线网状网比纯的自组织网络更加结构化和稳定，由固定和移动混合的客户端和路由器组成。网状网的路由器构成了用于为其他节点转发数据的骨干网[24]。它们还能提供连接其他网络的网关功能。

网状网是一种有望用于大规模、泛在性的高速率通信网络的解决方案。然而，要充分发挥网状网的优势不是一件容易的事。由于干扰的原因，大规模自组织网的容量是有限的，经过一长串路由器的通信吞吐量会迅速下降[25]。改进网状网性能的方法包括同时使用多个无线接口，波束形成天线，多输入多输出技术（MIMO）和机会信道选择，所有这些技术都致力于增加通信的差异化，从而获得更大的容量。

在 IEEE 中，不同的工作组都关注网状网。IEEE 802.11s 的目标是优化和扩展 IEEE 802.11，从而使它能够支持自组织和网状网。在 MAC 层功能中增加了网状拓扑发现和路径选择，并通过增加新的功能使多跳组网时的媒介访问更为有效。安全功能和网状测量也是 IEEE 802.11s 的一部分。在 IEEE 802.16a 中也有针对 MAN 的类似的工作，IEEE 802.15.5 则针对 PAN。IETF 的工作则主要面向自组织和网状网的自动配置功能。

网状网的开发工作与标准化工作并行进行，如麻省理工学院的基于非私有方案的 Roofnet 项目以及一些公司提出的私有方案（如 Locust World、Tropos、BelAir 和 Firetide）。据《MuniWireless》杂志（http://www.muniwireless.com）介绍，到目前为止，美国已经部署了 50 多个城市或乡村的网状网，其他国家亦在跟进。

如果网状网的主要目的是用于自组织网的互联，并且有一个或多个接入到固定网络基础设施（如互联网）的接入点，则通常被称为“多跳无线接入网”。在这样的网络中，有些设备有直接到接入点的连接，从而能够为那些没有接入点连接的节点来中继数据。网络拓扑通常组成一个树，接入点作为根节点。通过这种方式，接入点的无线覆盖范围可以延伸到更大的区域。

2.4.3　传感器网络

传感器无线连接的使用催生了很多新的应用并超越了“传统”通信的领域。传感器网络的主要应用领域是环境监测，传感器被放置在需要进行观察的地域，每个传感器都有一定的感知任务，如测量温度或湿度，记录声音和检测振动。测量的数据被加以分析并共享给其他传感器，最有价值的数据被转发给专用的接收节点，接收节点可以通过远程访问。

应用领域包括：

1）生物（如动物观察、冰川监测、海水监测和葡萄监测）。

2）市政和工业工程（如智能建筑、桥梁监测和自动化）。

3）医药和健康护理（如病患监护）。

4）紧急响应和军事（如灾难告警和车辆跟踪）。

为了易于部署，传感器设备应该是廉价且很小的，有较长的使用周期，这些对于开发高效的软件和硬件解决方案是非常重要的。基于这个原因，传感器的协议设计需要特别谨慎，从而能够有效地利用有限的能量、计算和存储资源。这些限制将来很可能会一直都存在，因为人们更倾向于改进技术来开发更小、更节能的设备，而不是使它们更强大。

无线传感器网络的技术方案与特定的应用相关且有非常广范围的实现方案。表2-1所提到的无线个域网的标准也适用于传感器网络。

IEEE 802.15.4是一种可用于低速率个域应用的无线技术。它的优点是低电量消耗和低设备成本。在WLAN中，设备传输使用无需牌照的2.4G Hz频段ISM。这个频段上的物理层定义了16个信道，每一个信道的数据传输速率为250kbit/s。在一些国家，还有附加的ISM频段868MHz（欧洲）和915MHz（美国和澳大利亚）可供使用。它们提供了10个40kbit/s的信道和1个20kbit/s的信道。所有的频段都使用直序扩频（DSSS）。媒介的访问控制使用载波侦听多路访问/冲突避免控制协议（CSMA/CA），可以使用分槽或不分槽的模式。拓扑形式是自配置的，拓扑结构可以采用带有协调节点的一跳星形拓扑，也可以采用多跳的对等（Peer-To-Peer）拓扑。基于IEEE 802.14.5，工业组织ZigBee定义了一组高层协议。ZigBee组网层包含了加入和退出网络的机制，并能在节点之间发现和维护路由表，且在组网层上定义了应用层框架。它执行协调节点的选择、发现节点所提供的服务以及其他一些功能。不仅于此，ZigBee还定义了MAC、网络层、应用层的安全服务。基于IEEE 802.14.5/ZigBee的商用产品包括Crossbow技术的基于MICAz平台的传感器和Ember公司的EM无线芯片。

另一种用于传感器网络的无线技术是超宽带（Ultra-Wide-Band，UWB）。信令使用非常短的脉冲来传递，从而占用了非常宽的频谱。尽管这些频谱与其他无线技术的频谱相重叠，接收器仍然能够从非常宽的扩频中监测并解码信号。UWB的主要优势在于：它能以非常低的功率进行传输；在短距离内数据传输速率非常高；传输能够有效地对抗选择性衰落。

还有一些私有的无线技术方案由一些公司和学术机构研发。在某些产品中，已经使用了蓝牙作为传感器网络的传输技术。一个著名的例子是由ETH Zürich开发的“BTnodes”。非无线传输包括使用光和超声波通信，后者特别适用于水下的应用场景，因为在那种环境下无线电通信不能使用。

从组网的观点来看，传感器组网与自组织的计算网络有些相似，如无中心的

控制和可能存在多跳通信的需求。一个主要的不同点是，传感器网络是面向数据的，即所操作的是在网络中穿行的测量结果数据。由于数据通常由一个传感器传向另一个传感器或数据分配节点，传感器路由协议通常使用树形拓扑，以数据分配节点作为根。因为传感器网络的容量限制和传感器节点的能量约束，聚合和压缩所经感知得到的数据是非常必要的。通过本地计算来减少数据的传输能够保存一些稀缺的资源。

2.4.4　可穿戴计算

可穿戴计算是指在人的身体上穿戴小型的、轻型的计算设备，如眼镜和衣服。穿戴的设备与用户进行交互，并能够使用用户的上下文信息和环境信息。可穿戴计算机在衣服或腰带中集成了主单元，与用户友好的输入设备（如手套、照相机、传声器等）和输出设备（如头戴式显示器、耳机等）相连，如图 2-12 所示。



图 2-12　集成在腰带上并附有头戴显示设备的可穿戴计算机

可穿戴计算的一个重要特性是用户所穿戴的计算机的所有部件或不同的穿戴计算机之间的通信。原则上，我们可以使用现有的技术来实现个人和局部范围内的通信。然而对于一些实时多媒体应用，现在的无线技术不能提供足够高的数据传输速率。基于这个原因，目前正在开发一些非无线的替代技术。如通过人的皮肤来作为传输信号的介质。人体周围的电场可以作为一种传输连接，且具有很高的数据传输速率。基于该传输原理的技术还在开发当中，一个著名的例子是由 NTT 开发的 RedTacton 系统。

可穿戴计算的应用范围非常广泛。在某些领域，这种方式与笔记本和掌上电脑相比有非常显著的优点。一个典型的例子是紧急响应，如可以让消防队成员装备可穿戴计算设备，头戴式显示设备和照相机。这种穿戴的计算和处理使他们之间能够协调工作，发送警告信息，感知环境的变化。另外一些应用场景包括工业生产、维护保养、医药和健康护理等。在消费领域，可穿戴计算和通信将催生新的娱乐形式、通信、导航和捕获、存档个人体验等。可穿戴计算的“轻量级”

版本现在已经广泛使用了，如带摄像头的手机、MP3 播放器等。

2.4.5 车载网络组网

在汽车中集成无线通信技术为车载通信提供新的安全和通信特性，如无线自组织网络能够使事故车辆向其他车辆发出事故告警信息，从而避免连环事故。其他安全服务包括基于传感器的交通拥塞警告。最后，还可以通过自组织网络为汽车中的人提供人与人之间的通信（如文本信息、游戏社区以及逐跳的电话通信）。

在该领域中，正在进行一些开发和标准化的工作。所有无线通信的基础是无线频率的分配，上面提到的与安全相关的车载通信不能被其他无线通信所干扰，从而需要分配专用的频率。在北美和日本，分配了 5GHz 的频带，但在世界上的其他地方尚未分配相应的频率。为了定义这种专用的，短中程通信的物理层和 MAC 层，IEEE 802.11p 工作组增强了 IEEE 802.11，使其适合于车载环境。另外为了推进这方面的工作还成立了一些工业联盟，如欧洲的“汽车-2-汽车通信联盟”。网络层的问题包括当大量的汽车集中在一个区域时，如何将消息路由都特定应用。这种情况下，可以使用 2.4.1 节所提到的基于地理的路由，利用 GPS 信息作为汽车的位置信息。

2.4.6 周边环境组网

泛在计算网络所面临的一个重大挑战是如何将不同的动态的无线网络自动地互连起来。一个典型的例子是如图 2-13 所示的一个个域网（PAN）、一个铁路网络和一个 WLAN 热点之间的互连。

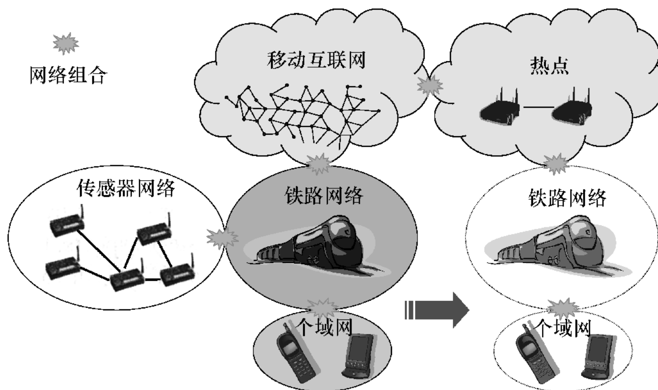


图 2-13　周边环境网络场景和网络组成

欧盟项目“周边环境网络（Ambient Networks）”将这个挑战作为一个问题

来研究，提出了一个动态的、即时的网络组合的通用框架[27]。网络组合的范畴超出了今天的互联网和蜂窝网络。网络的工作不仅是基本的编址和路由这个层面，还需要增加一些附加功能，如在不同的网络中各使用不同的移动性处理方式。由于过程复杂且需要即时失效，并保证对用户透明，这种网络的控制面配置需要自动进行。

2.5　可编程网络

设计一个高效可靠的通信协议需要付出很多的努力，且通常由于标准化的进程而拖长。因此，引入一个革新通常需要较长的时间。一种更具灵活性，能缩短推向市场时间的主要技术是可编程网络。可编程网络的节点之所有能够快速地部署新服务是因为对网络资源使用了开放接口。通过这项技术，可以在网元上安装各种新的协议，使用底层的资源来生成新的服务。

在设计网络系统和协议时，后向兼容是一个重要问题。然而，现有的网络在扩展性和与未来技术的无缝连接上不够理想。一般情况下，现有网络都需要升级设备的固件才能支持新的软件版本或对软件补丁进行修正。

基于这样的情况，有人认为应该在应用层和中间件上增加更多的灵活性，而在网络层上则可以要求低些。让软件在运行时具备一定的自适应性是目前在许多应用中广泛采用的一种技术，如在媒体播放器中动态地安装编解码器。类似的情况也出现在网络协议设计中，只有基础协议或机制是必须的，而其他扩展则按需配置。

对于移动网络，我们认为同样需要灵活性，移动网络因为无线连接、移动性而显得非常脆弱，优化更容易取得成效。而且异构网络中新的空中接口技术的创新也需要网络不断适应新的技术。

2.5.1　适应性的概念

下面我们将讨论适应性概念在通信设备中应用日渐广泛的趋势。长期以来，在各个领域，技术都在朝着可配置、可编程的通用平台的方向发展，灵活的适应性正是顺应这个方向的。半导体技术和软件技术的发展为这个技术方向的发展提供了支持，例如在中间件领域，类似 Java 这样的执行环境已经日渐发展成熟。对于底层网络技术，如网络处理器、无线码片集等，提高它们的可编程的程度是当前的趋势。

我们在本书中提到的可编程的概念比较宽泛，从而能够覆盖系统从可重配置的无线参数到应用的各个方面。由于范围宽，所以涉及各种不同概念的适应性。根据可配置性和可编程性，可以将所支持的适应性分成以下类别：

1）配置：无论是在系统启动前（如在开发过程中或配置过程中）还是运行中。

2）软件的参数化：这是一种常用的，无需升级软件而能够引入灵活性的方式。

3）完整的软件升级或更换：典型的例子如固件的升级，这经常会导致设备或服务器在一定的时间内不可用。

4）复杂软件系统中的部分软件升级：由于没有开放的接口，系统的行为常常需要重新评估，可能会导致系统中断或其他后果。

5）在开放式平台上安装组件，通过接口的设计来确保功能的正确性和无缝的服务演进。

6）在数据交换的过程中自动安装软件，并通过安装的软件直接影响到所交换的数据，这通常称为主动网络封装的方法。

针对上述列表，我们关注于以下开放平台，当然，其他形式在实际应用中也是很重要的。最后一种方法，针对代码执行的位置提供了很好的灵活性，但是带来了很大的安全风险。

尽管适应性是一个重要的概念，但我们还是应该注意到它所带来的隐含的问题。在很多情况下，附加的灵活性会带来过高的启动成本。然而，如果系统缺乏灵活性，需要整个被替换时，价格也是非常高的。成本的主要因素来自于比最初实际所需的要贵得多的硬件和对一个可靠的可编程环境的开发。对于软件而言，灵活性的需求将导致更多的变量和中间层，从而需要更加复杂的设计。但是，在大型应用中可编程的物理参数或灵活的软件平台将使系统能够无缝地演进，从而会更加经济。

下面，我们将介绍开放平台的概念[27]以及它和系统架构、跨层接口的关系。一个抽象层可能会有多个提供实际计算工作的开放平台组成，每一个这样的平台都包含一个稳定的、最小的基础平台并在其上增加一些平台组件，这些组件根据时间要求，也能够被添加或者删除。

2.5.2 可编程网络基础设施节点

移动网络网元的通用架构如图2-14所示，在这个架构中，我们考虑3个平台，每一个都是可编程的，包含一些可配置的组件。

1）计算平台是一个多功能的平台，处理包括路由、QoS信令和连接管理等功能在内的有状态的协议。

2）转发引擎是网络节点的数据通路，用以连接网络接口平台，如一个交换矩阵，该引擎可以由专用硬件实现或通过操作系统的核心模块来实现。转发引擎被编程用于关键性能的任务，需要逐包地进行数据处理。

3）网络接口是不同无线或有线标准之间的媒介。它们被配置成或编程为可以适配不同的物理协议并向高层触发事件。

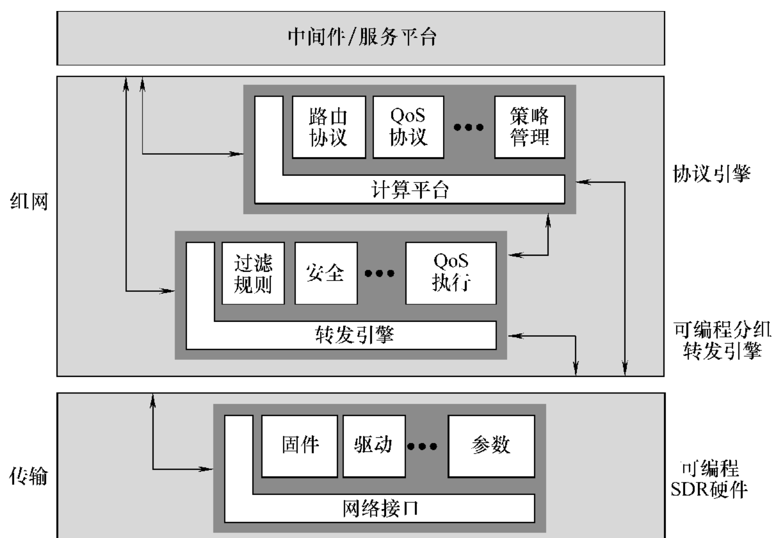


图 2-14　可编程网络设备架构（《Communications and Networks》已授权）

从具体操作的观点来看，这些平台的需求如下：

1）提供不中断服务的可靠性，特别是在服务升级时。

2）远程管理，这点对大网络的集中配置非常重要。

3）对抗外部攻击和风险的安全特性，由于我们假定网络是由运营商运营的，主要的安全风险来自于外部接口。

我们假定在中间件抽象层有一个配置管理器，可以在更低层安装新的组件。由于配置可以远程执行，分布式处理平台比较适合于这项任务，并且可以为复杂的配置情况提供事务处理服务。

2.5.3　可编程的移动终端架构

终端侧的可编程架构如图 2-15 所示。

1）智能卡网络平台，提供如编址、认证之类的网络功能。

2）智能卡中间件平台，提供签约用户标识和高安全特性的执行环境。

3）可编程的无线电平台，可用于一种或多种无线电标准族。

4）一个原生的操作系统平台，对协议栈和多媒体编解码等关键应用提供实时操作支持。

从运营商和制造商的观点来看，主要的需求包括：

1）可生存性，如对于误配置、故障或误用等情况下的鲁棒性。

2）针对终端用户、运营商、制造商的安全性。如设备商关注是否存在因设备问题而引起的不正常的无线电发射。运营商关注可靠的服务和计费。用户关注

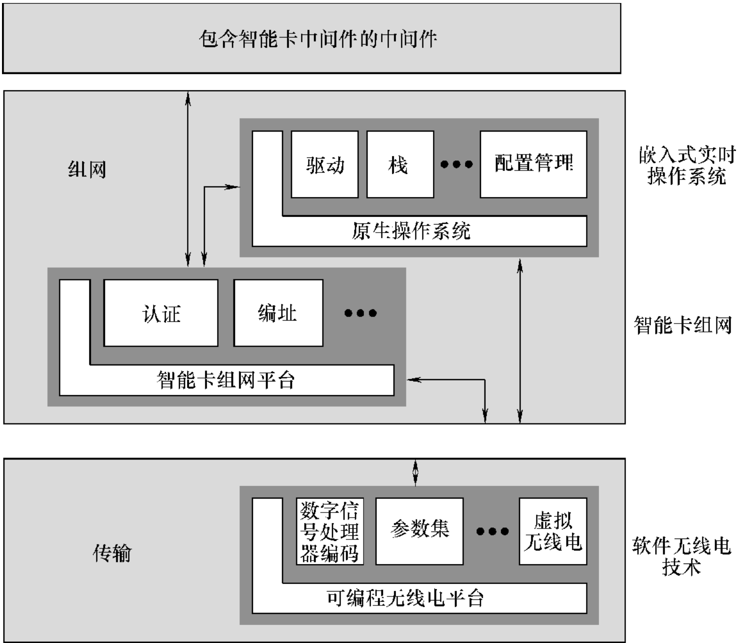


图2-15 可编程的网络设备架构（《Communications and Networks》已授权）

设备的完整性和服务的可用性。

3）硬件、软件平台的上市时间和针对大众市场的优化手段以及灵活性或升级方面的需求。

可编程网络中有一种比较显著的技术是主动网络，网络节点被编程为可以针对经过它们的数据包执行定制的操作。主动网络技术提供了一种与硬件和操作系统无关的执行环境。主动网络的方案包括 Decasper[29]、Wetherall[30]等人提出的。其中一些方法的设计目标是为了获取灵活的、逐包处理过程的高性能，有些则是为了获得控制面的灵活性。

2.6 小结

无线通信技术正在飞速发展，并应用于越来越多的领域。未来的蜂窝网络将连接到各种不同技术或直接在不同的技术上运转。实现这些通信领域的扩展需要一系列的支撑技术，如高效的 QoS 和切换机制。

在本章中，我们简要介绍了与4G网络相关的无线技术，其中一些正在标准化（如 WLAN、WiMAX 和网状网）并将直接用于4G，而另外一些（如可穿戴的计算和可编程网络）则有待于进一步的研究。

第 3 章　移动服务系统

Wolfgang Kellerer

成功的下一代移动通信系统预计将会把其焦点放在用户和应用上。服务平台有提供了创造新应用的能力。泛在服务平台为服务提供带来了新的机会。

本章将介绍移动通信环境中最新出现的支持服务提供的服务平台。我们不仅仅关注蜂窝移动通信系统，而且像前面章节一样，还关注于整个泛在通信环境。泛在服务平台使得各种异构设备能使用同一个服务，或者像 P2P 系统一样高分布化。我们认为这样的泛在通信将作为运营商服务环境的主要发展趋势之一。

我们首先对服务平台进行简单的介绍，以对已有平台有一个整体的了解。其中包括基于 IP 的平台，如 IP 多媒体子系统（IMS）；非 IP 平台，如移动增强逻辑的客户化应用（CAMEL）。此外还包括开放 API，如开放系统接入（Open System Access，OSA）；移动互联网平台，如 WAP（Wireless Application Protocol）以及 i-mode 服务。

3.2 节将讨论下一代服务平台带来的挑战和应具备的能力。扩展传统运营商服务平台到支持传感器或者 P2P 系统的泛在服务平台就是新出现的挑战之一。这里只简单介绍未来服务所具有的能力和特征，并将在后面的章节中详细讨论。

3.3 节将利用一个具体示例来介绍泛在服务，主要包括在异构多设备环境中的无缝服务连续性，以及基于 IP 服务平台的具体实现。IP 服务平台主要用 SIP 来实现多媒体信令。

3.1　服务平台一览

自从模拟电话在世界范围内普及以来，通信服务的前景就发生了翻天覆地的变化。因为之前的通信系统只支持一种服务，所以它没有独立的服务平台。随着日益增长的应用需求，对服务平台的要求也越来越多。服务平台通过为异构的通信系统建立一个抽象层，以便能为用户提供不同种类的服务。互联网与电信网的融合，在给服务提供带来新的机遇的同时，对网络的可靠性和开放性也提出更高的要求。目前不仅运营商能提供服务，还有大批涌现的第三方也通过运营商平台提供服务。而且，运营商之间的互操作不应仅局限于语音漫游服务，还应包括世界范围内所有可接入的服务。

为了使读者能更好地理解服务平台架构的演进，我们首先对已有的服务平台从电信通信中的智能网到IMS和移动智能网做一个概述。

3.1.1 移动服务和支撑平台

移动服务，如移动电话极大地影响了我们的生活，在通信和信息领域几乎没有其他任何词语可以像移动服务一样高频率地被提到。因此，为了使读者能更深入地理解服务平台和架构，我们首先给出本章所用术语的定义。

在信息通信系统中，服务是由供应商提供，并控制提供给用户的功能实体，服务依靠纯粹的系统能力来满足用户的特殊需求，并通过专用接口为用户提供服务。服务还可以是其他服务的用户。在电信通信系统中，服务是以具体方式为用户实现信息交流的实体。在这里，连接的建立和释放跟用户、媒体的添加和删除一样，都是基本元素。视频会议、呼叫转移、即时信息和电子邮件都属于服务。

移动服务通过移动通信系统来交换信息，无线通信系统的路径中至少有一条使用无线连接。

应用通过给用户提供一个特定的用户界面或者特定信息来完成对用户的特定服务。举个例子，在移动互联网中的WAP系统就是一种移动应用。用户使用WAP浏览器通过无线连接享受接入互联网服务（如获取数据），WAP浏览器就是运行在用户手机上，由供应商提供的门户网站地址。

服务平台是系统中的支撑服务组件，图3-1所示为典型的服务组件和融合这些特点的特定服务的服务逻辑。

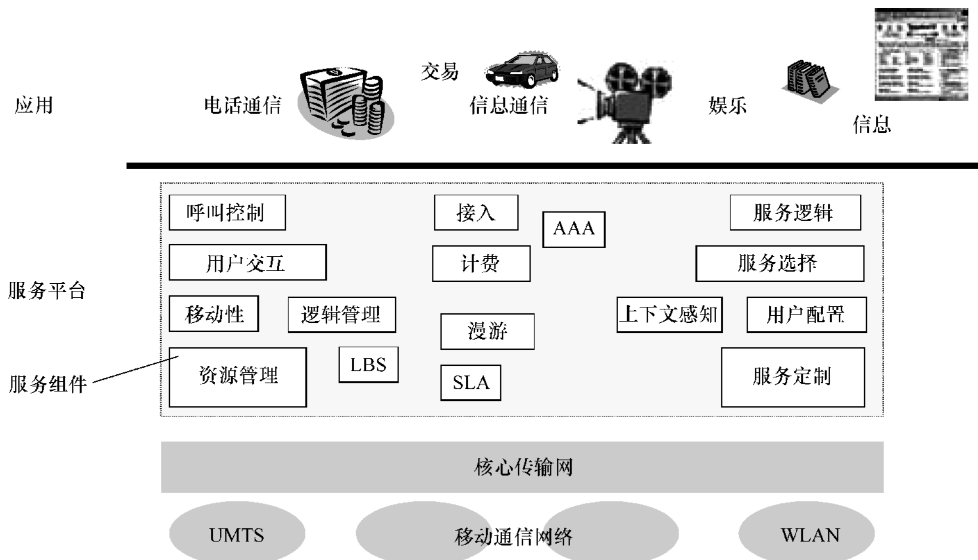


图3-1 服务平台的功能

以下是服务平台具有的功能：

1）控制功能：呼叫控制、会话控制、用户交互、资源控制、QoS 控制。

2）接入功能：鉴权、认证、计费（AAA），发现、选择、计费。

3）增值功能：定位、上下文感知、用户配置。

4）移动性支持：漫游、服务等级协议（SLA）。

下面介绍几种在移动通信系统中使用的服务场景。

1）个性化：旅行意味着总是要安排很多事情。使用个性化服务，用户可以在同一时间放心地选择和协调多项事情。例如，商务旅客在租车参加某一会议时并不需要输入参会议的时间和目的地到导航系统，也不需要在车载显示器上调整信息服务。此外，还可以使行车路线经过他所在银行的 ATM 自动地预定商业晚餐，甚至还可以考虑用户的偏好。

2）泛在服务移动性：用户在去他同事办公室的途中，也许正通过手机参加视频电话会议。一旦他到达同事办公室时，他的位置可以立即被环境注册，通信设备也能发现并反馈给他。如果想要得到更好的视频会议质量，用户还可以将正在进行的视频会议的音频流转接到房间内的音响设备，而视频画面则转接到电视屏幕。

3）泛在 P2P 服务：用户可能会在歌剧即将开始前突然想卖掉门票，这时她手机上的移动 P2P 应用或许会派上大用场。用户将门票的信息发布出来，其他人则以 P2P 的方式根据其兴趣进行搜索。这样不用奔波于每个订票点，交易（包括谈判和付款）就直接在系统中进行。

互联网系统由分布式实体组成，如终端、网络路由器和网络服务器等。如图 3-2 所示，从服务平台架构的不同概念中，根据其主要控制模块所处的位置，我们可以将服务平台架构原理分为几大类。早期的多服务平台，比如综合服务数字网（Integrated Services Digital Network，ISDN）是将服务集成到用户网络接口（User Network Interface，UNI）。这样有个缺点，当有新服务引进时，所有的设备和交换机都必须更新。集中式的服务平台架构（如智能网，IN）主要利用一个中心服务器来存储服务，这样可以更容易实现和删除，但是可扩展性差。互联网通常提供端到端的服务，比如终端与终端之间或者终端与服务器之间的服务。P2P 服务的情况更是极端，因为它的企业平台只由用户设备构成。20 世纪 90 年代中期，如电信信息网络（TINA）之类的架构得到演进，它们通过中间件平台互联异构服务平台组件来隐藏底层系统实体，而现在的架构是这些概念的融合。

3.1.2 电信服务平台

如前所述，服务平台已由先前的单服务平台发展为多服务平台，如图 3-3 所

示。我们首先了解固网的服务平台。

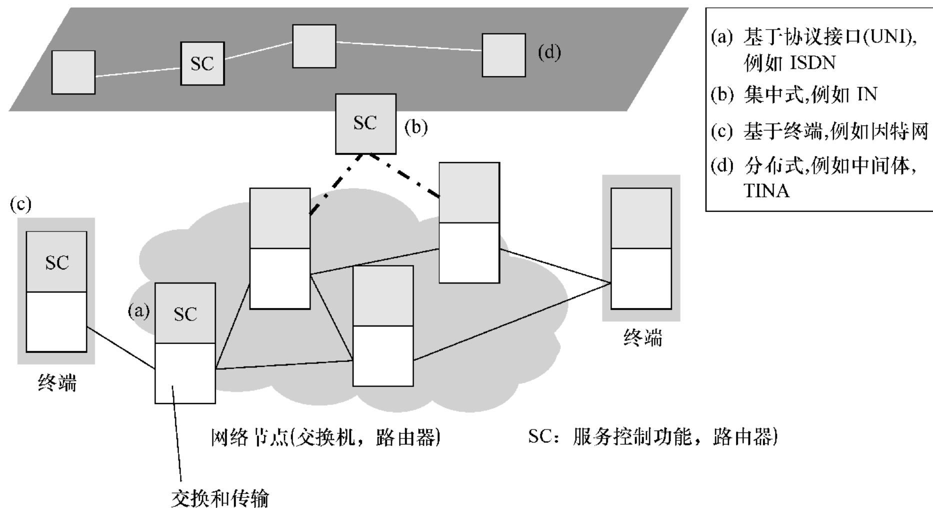


图 3-2 服务控制分布

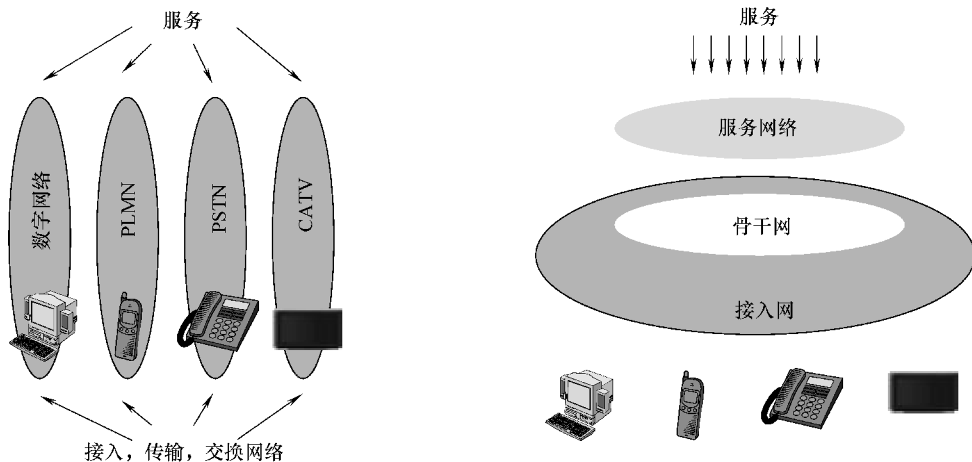


图 3-3 服务架构演进：公众陆地移动通信网（PLMN），公共交换电话网（PSTN）和有线电视网（CATV）

在 ISDN 系统中，由第三层协议中的 D 信道（Q. 931）来完成服务控制，包括基本服务中的基本呼叫控制，如语言、传真、基于电路交换链路的数据服务以及 ISDN 功能的初始化。Q. 931 是用分散的方法进行服务控制，因为它运行在所

有的交换节点上，这就意味着当引入新服务时，所有交换节点的协议软件都必须更新。这种方法是基于封闭网络的，所有的资源都由运营商提供和控制。

智能网是一种为像 ISDN 这种以协议为基础、分布式的网络提供增值服务的架构。在传统的系统中，服务控制功能直接由交换节点来完成，如图 3-4 所示，智能网中存在一个中心控制服务器，即服务控制点（Service Control Point，SCP），它通过一个单独的信令协议与其他所有具有信令协议的交换节点通信。通常用 No. 7 信令来完成基本呼叫流程和信息传输的信令部分。智能网应用部分（INAP）定义了服务信令协议消息，它是 No. 7 信令系统的可选部分。在这种方式下，INAP 消息集就决定了服务控制的能力。智能网出现在很多服务平台架构中，比如我们下节将会介绍的移动系统。对于运营商来说，引入智能网的主要动机是当有新服务加入时，可以不依赖于交换系统设备商，由第三方提供服务并不是主要目标。

分布式处理环境（Distributed Processing Environment，DPE）支持在异构平台上进行分布式开发，而不用进行应用层的适配，比如基于中间件的 CORBA[5]。DPE 中包含了在系统组件中交互必要信息的机制，比如远程过程调用。

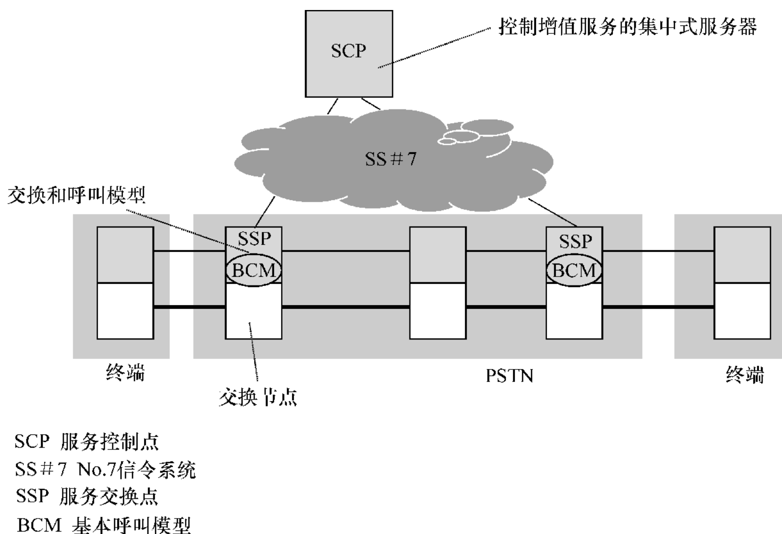


图 3-4　智能网

电信信息网络架构（Telecommunication Information Network Architecture，TINA）[6]基于面向对象的方法，为多媒体信息和通信服务定义了高级服务、网络以及管理架构（见图 3-5）。TINA 系统组件通过 TINA DPE 进行交互。这种方法使得 TINA 服务平台体系架构只需要在组件接口中描述即可。

TINA 中最重要的概念是会话，例如，交互式会话中的服务控制功能模型。

TINA 对接入会话、服务会话和通信会话进行了区分，每一种会话都需要服务器来完成相应的会话控制，例如，在接入会话中会有用户的认证和鉴权。

图 3-5　TINA 整体架构

如图 3-6 所示，TINA 服务体系架构详细说明了会话中会相互影响的组件和构建 TINA 服务平台所需要的组件。接入会话定义了用户如何接入平台以及平台中的服务。运行于用户设备中的供应商代理模块与服务平台侧相应的用户代理模块相连接。服务平台侧的用户代理代表了用户，并进行计费控制和服务选择。一旦用户选择了一种服务，服务供应商就会针对这一服务实例化相应的服务控制组件。服务会话管理器（Service Session Manager，SSM）是主要的控制实体，用户会话管理器（User Session Manager，USM）针对每一个加入的用户，而特殊服务实体（Special Service Object，SSO）则作为可选实体，例如，在需要时可创建内容服务器。用户侧的服务会话用户应用实现了用户接口。在需要时，通过用户代

图 3-6　TINA 服务架构

理模块可以邀请更多的用户加入会话，并初始化相应的组件。作为通信会话的一部分，SSM 与通信会话管理（Communication Session Manager，CSM）之间的接口用于建立参加实体之间的连接。

TINA 激发了其他几种服务平台或平台组件的演进，例如，在 Parlay 将接入和服务会话相分离。然而 TINA 至今未得到商用，其主要原因在于它的早期开发没有考虑当前的 IP 技术，而是基于异步传输模式（ATM）；再者，从系统演进的角度来看，TINA 早先在具体实现时，并没有考虑与现有系统的互通。

Parlay 系统使得服务平台向其他服务提供者开放，它提供了一个标准的应用程序接口（Application Programming Interface，API），第三方服务提供者通过 API 接入运营商服务平台，移动通信网络也使用了 Parlay API，这里被称为开放服务体系架构（Open Service Architecture，OSA），详细介绍请见 3.1.5 节。

3.1.3　蜂窝网络服务平台

受到固定电信网络体系架构的影响，移动网络服务平台体系架构也从语音系统向开放服务平台演进。另外，我们可以观察到基于分组传输互联网模式是如何集成的，以及它是如何影响服务平台体系架构的。

如图 3-7 所示，欧洲的第二代移动通信系统 GSM（Global System Mobile Communication，全球移动通信系统）的服务平台主要由归属位置寄存器（Home Location Register，HLR）和主要负责基本呼叫控制的移动服务交换中心（Mobile-service Switching Center，MSC）组成。HLR 的功能与智能网体系架构中的 SCP 类似，用于存储用户数据，例如用于移动性控制的用户当前位置信息和额外的增值服务信息。HLR 服务集中与用户相关的服务，例如在基本呼叫控制中由触发器调用的忙时

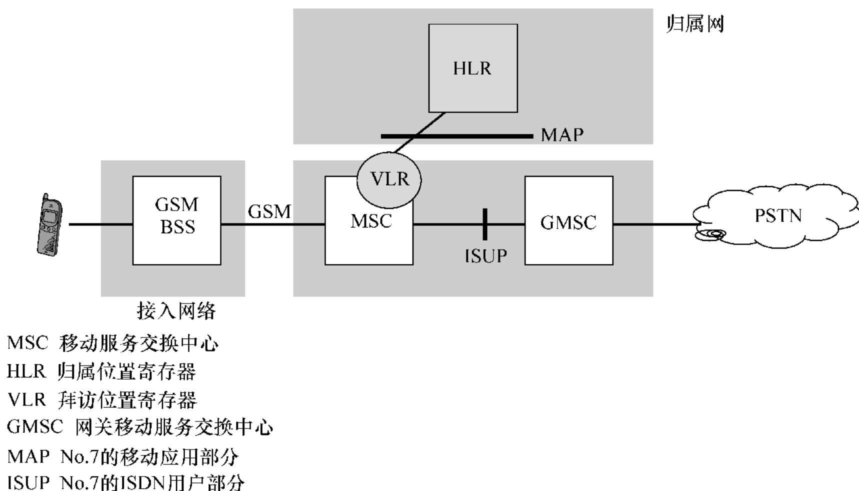


图 3-7　GSM 服务体系架构

呼叫转移。对于像号码转换之类的全球增值服务则由带有中心服务器的CAMEL系统来完成。

CAMEL是在GSM系统中构建的高级智能叠加网。除了在GSM HLR中用于个人服务的服务控制点外，CAMEL体系架构还提供了新的服务控制点（gsmSCF），用于全球服务，与交换节点中的基本呼叫控制分开。图3-8所示为GSM系统中CAMEL体系架构，gsmSSF上的服务交换功能用于触发增值服务呼叫。

从服务平台的角度来看，2G到3G移动系统的演进主要是由于引入了基于分组传送方式。GPRS（通用分组无线服务），我们通常提到的2.5G，就是在当前的电路交换传输中引入了一个新的域——GPRS。GPRS引入了新的组件，如SGSN（服务GPRS支持节点）和GGSN（网关GPRS支持节点）。SGSN实现在移动通信系统中的IP通信，而GGSN则实现了全球互联网的交互。

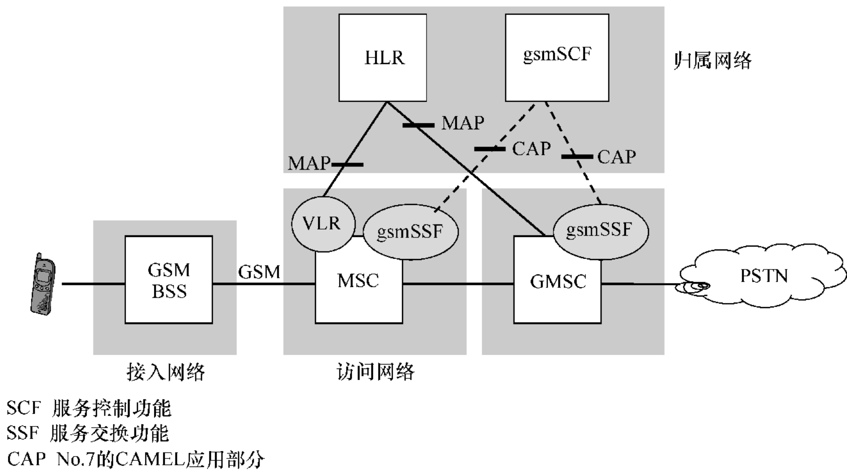


图3-8 CAMEL

UMTS由欧洲提出的作为第三代移动通信技术，包含了电路域和分组域，提供基于电路交换的语音通信和基于分组交换的互联网连接。此外，如图3-9所示，UMTS中引入IMS（IP多媒体子系统）来控制IP多媒体服务，在下面章节中，我们将介绍IMS服务平台。

IMS建立在分组交换域上，HLR/CAMEL则由GSM演进到电路交换域，UMTS包含了由虚拟归属环境（Virtual Home Environment，VHE）所描述的高级服务体系架构组件。VHE是个人服务环境中的一个概念，它存储了UMTS的用户个人设置（用户配置数据、用户偏好、个人服务等）。VHE中阐述的个人服务环境的需求在于用户跨网络和终端之间的可移植性和可接入性。VHE概念的目标在于用户在访问网络时可以无缝地使用其归属网络中的个人服务和配置数据，

VHE 概念它为每一个移动系统组件，如 SIM 卡、终端和网络都指定了一个通用的服务处理和数据存储点架构。

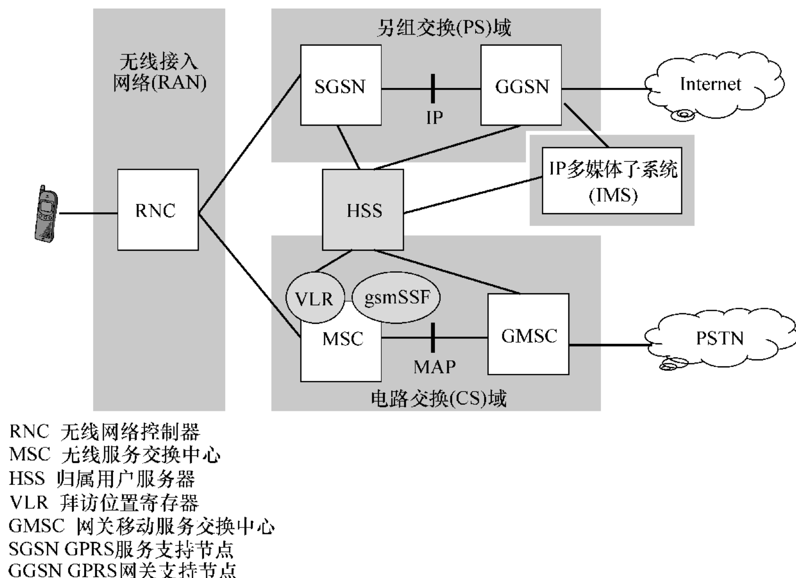


图 3-9　UMTS 服务体系架构

在电路交换域，CAMEL 就是 VHE 概念的一种实现，因为 CAMEL 使用基于智能网的技术，当用户漫游时也能使用 CAMLE 服务。然而这对于语音服务却有所限制，而 CAMEL 服务平台的目标正是语音服务。在 UMTS 中实现还需要的组件包括移动应用处理环境（Mobile Application Execution Environment，MExE）和 UMTS SIM 应用工具箱（UMTS SIM Application Toolkit，USAT）（请见第 5 章）。它们分别定义了在移动终端和在 SIM 卡中处理服务的环境，各自的下载机制和安全功能使得这些环境成为移动手机中处理个人服务的可靠平台。

3.1.4　基于 IP 的移动服务平台

互联网上的服务是基于端到端的原则，从而解决了网络服务器的复杂性问题，也使得创建新的仅基于终端软件的服务和应用更加简单，因为它们仅仅基于终端软件。互联网服务平台主要特点是一些用于控制会话的协议。这里的会话指的是一系列有一个或者多个数据连接支持端到端的有意义的通信。数据传输可以与会话信令有关［例如，超文本传输协议，（Hyper Text Transfer Protocol，HTTP）］，也可以独立于会话信令［例如在一个流会话中，用于信令的实时控制协议（Real Time Control Protocol，RTCP），用于传输的实时协议（Real Time Protocol，RTP）］。

对于会话信令，关于互联网上的多媒体服务控制的两大主要协议已经实现了

标准化。ITU-T标准化的H.323协议簇（在1996年12月第一次发布），被认为是一个没有局域网服务质量保证的实时视频会议标准。IETF多方多媒体会话控制工作组标准化的会话初始化协议（Session Initiation Protocol，SIP），是一个IETF多媒体服务平台的基本协议，并在1999年第一次提出。上述两个协议提供了相似的功能，并且在设计上有比VoIP更广的用途。H.323和标准化的H.450，为了取代VoIP的私人电话交换，特别注重电信增值服务和电信网的平滑互联。一开始，SIP有较广的范围，也重视非VoIP服务（例如即时信息），但SIP仅仅定义了必需的与服务特征相关的协议消息。Glasmann给出了两个协议簇的服务结构的更多内容。对于IP上的会话信令，通用移动通信系统标准化组织3GPP已经选择了SIP作为IMS的基本协议。

1. SIP

SIP服务平台的架构由基本的SIP及其扩展协议构成，并运行在实现了SIP用户代理或者网络服务器（例如SIP代理服务器）的用户终端上。此外，附加服务可以在代理服务器或者实现了SIP的SIP用户代理上实现。到目前为止，已经有好几种脚本语言可用于SIP服务的开发，例如呼叫处理语言（CPL）可用来开发针对SIP代理上的服务。总之，SIP服务平台由多层构成，如图3-10所示，其架构可用于实现诸如即时消息服务、语音服务、语音增值服务等。SIP核心协议是SIP服务平台的基础，支持大多数服务场景。SIP扩展协议是针对特定服务类型的通用功能。SIP实体上的服务脚本进一步定义了SIP逻辑。值得注意的是，IETF在设计SIP时，其目的是将SIP设计成一种不局限于语音功能的通用的会话初始化协议，因此在设计其所有的扩展协议也会做仔细的考虑。

SIP是IMS的基本协议，在3.3节介绍的具体系统实例中也使用了SIP，故在本节我们将详细地介绍该协议。为了更好地理解协议的概念，我们将提供更多的协议细节。关于会话的建立和控制，SIP提供了一系列基于文本的消息（被称为方法），这些消息在SIP对等实体间（在用户端的SIP用户代理）以独立事务的方式进行交互。REGISTER，INVITE和BYE方法是会话参与者必须完成的动作。每一个方法都由一方的请求消息和多条响应消息构成。例如，一个INVITE请求用于告知接收者将要建立一个多媒体会话，如果接收者同意建立会话，则会发送一个“200OK”的消息响应。会话请求者在接收到“200OK”消息之后会回复一个“ACK”确认消息，从而完成一个会话建立事务。会话中的媒体数据传输路径独立于基于SIP的信令路径。在信令传输路径上，一些网络实体（例如能被消息穿越的代理服务器和重定向服务器）可以完成地址解析等功能。图3-11描述了一个典型的SIP系统，称为SIP梯形。INVITE消息穿越了好几个代理服务器，而ACK消息和数据流则分别采用了独立的路径进行传输。

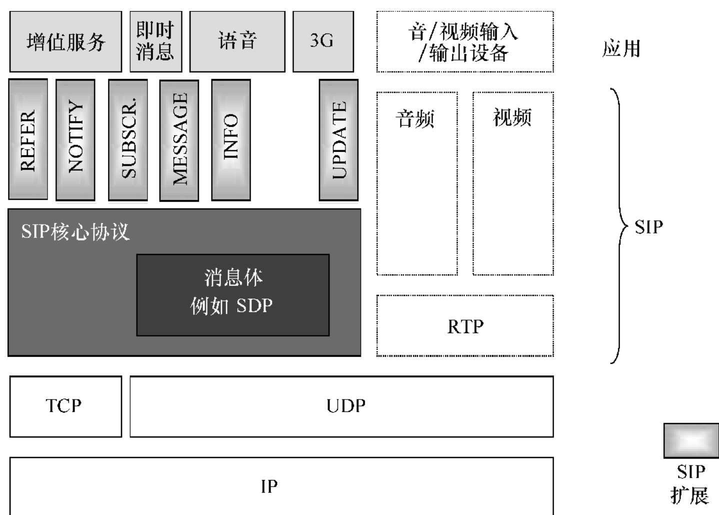


图 3-10　SIP 服务体系架构

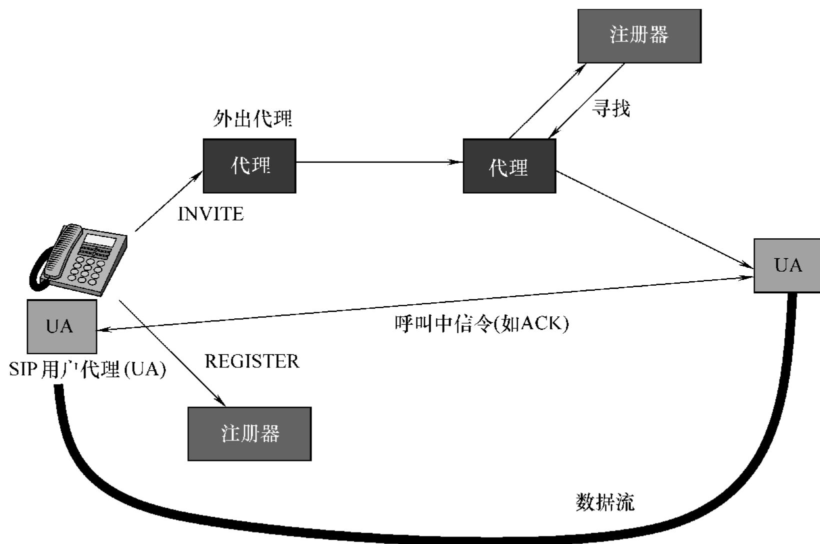


图 3-11　SIP 梯形

会话通过两个层面来加以描述。SIP 消息方法头中包含了会话参与方的地址和处理特征参数，而在多媒体会话中媒体流参数的协商则由另一个协议来完成。IETF 建议使用会话描述协议（Session Description Protocol，SDP）[16]。它事实上不是一个协议，而是一个结构化、基于文本的媒体描述，并承载于 SIP 消息体中。下面是一个 SIP 消息的例子。

```
INVITE:sip:bob@ docomo. com SIP/2. 0
Via:SIP/2. 0/UDP pc 12. company. com
To:Bob < sip:bob@ docomo. com >
From:Alice < sip:alice@ company. com >
Call- ID:a84b4c76e66710@ company. com
Cseq:1 INVITE
Contact: < sip:alice@ company. com >
Content- Type:application/sdp
Content- Length:153
V =0
o = alice 53655765 IN IP4 pc33. company. com
s = Session SDP
t =0 0
c =  IN IP4 pc33. company. com/127
m = audio 3456 RTP/AVP 0
m = vedio 51372 RTP/AVP 31
m = application 32416 udp wb
```

上面的例子给出了一个典型的 SIP INVITE 请求消息。首行内容指出本消息所用的方法、接收者的地址以及 SIP 版本号。该消息必须穿过发起域的 pc12 服务器（via：头），To 和 From 头分别指出消息的接收者和发起者。Contact 头是可选的，用来描述负责会话控制的一方，例如第三方呼叫建立。Call- ID 唯一地标识该会话，而 Cseq 在请求消息流中标记消息的顺序。最后，Content- Type 头和 Content- Length 头则表示了与之相关联的会话描述，在这里是 SDP。

SDP 描述以版本号、发起者的名字、会话标识、发起者的主机 IP 地址、会话名称和会话时间表（t =0，0 表示立即开始和无限会话时长）开头。发起者应用的连接数据（IP 地址和端口号）开始于 c 行，紧接着是媒体流描述，如媒体流类型（语音、视频或者诸如 white board/wb 之类的具体应用）、端口号和带有格式描述符的传输协议。

2. IMS

3GPP，IMT2000 家族的标准化组织，指定 IP 多媒体子系统（IMS）作为一个支持分组环境的多媒体服务的系统架构。IMS 允许运营商在基于互联网的规则下更灵活地创建和控制新服务。另外，IMS 提供给运营商更多的服务提供控制权，而不仅仅作为为移动用户获取数据提供互联网连接的管道，就像 GPRS 所做的那样。这主要是通过基于 SIP 的服务层和分组交换域的传输层相互关联实现的。SIP 中已经加入了运营商的一些需求，例如服务质量（QoS）控制、计费和订阅管理等。IMS 是 IMT2000 的面向分组域的一部分，同时也是对电路交换域的

一个补充。就像在未来的计划中所有的服务都将基于分组操作一样，IMS 也不会只限制到核心网。

图 3-12 给出了 3GPP 规范中 IMS 的基本组件。不同的线路表示了不同的信息路径：在电路交换域信令和数据的路径是分开的，而在分组交换域则是相同的路径。虚线表示信令信息。在左手边的接入网（AN）用无线网络子系统（RNS）来表示。中间是分组交换核心网（PS CN）域和电路交换核心网（CS CN）域，它们两者都和 HSS 交互，HSS 是 GSM HLR 的升级。IMS 域则位于图中的右边。基本上，3GPP 架构是 SIP 和 MEGACO 协议的结合体，其中 MEGACO 用于控制通过媒体网关的媒体流。UE 作为 SIP 用户代理，用来发起和终止 SIP 请求。IMS SIP 服务器有 3 种类型：代理呼叫会话协议控制功能（P-CSCF）问询 CSCF（I-CSCF）和服务 CSCF（S-CSCF）。应用服务器（AS）作为一个 SIP 端点，提供诸如媒体流的服务。CSCF 服务器控制数据流，在需要与传统语音网络互通时，则通过 IMS 媒体网关（IMS-MGW）和媒体网关控制功能（MGCF 和 BGCF）来实现。

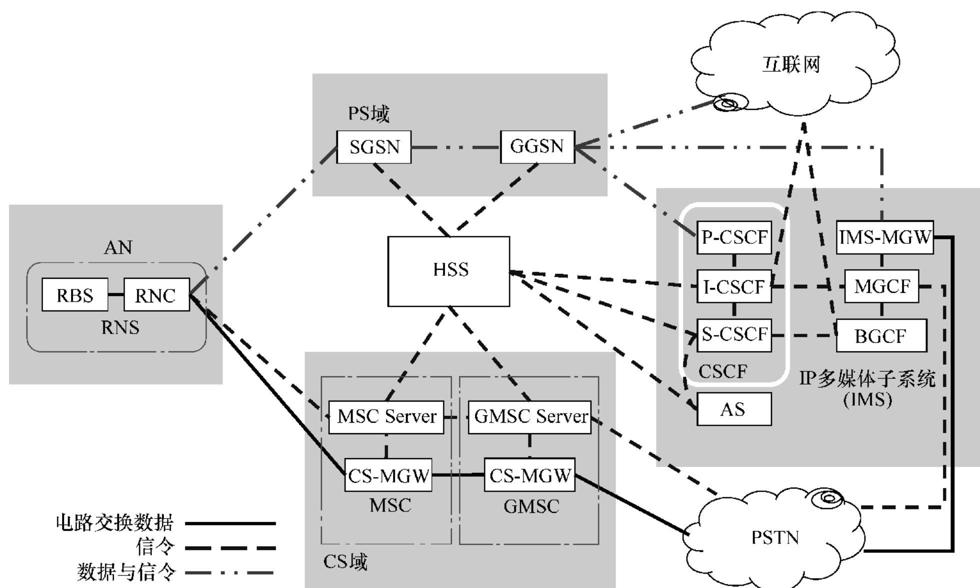


图 3-12　IMS 体系架构

P-CSCF 是 IMS 漫游用户的第一个联系点，它作为用户接入归属网络 IMS 域的代理。I-CSCF 是不同运营商的 IMS 域之间的联系点，它隐藏了 IMS 系统的配置、能力和拓扑结构，并为用户接入到归属域提供认证和鉴权。I-CSCF 知道用户所属的 HSS 以及用于处理用户服务的 S-CSCF。S-CSCF 的功能则是进行会话

控制和处理会话状态，它还能作为注册服务器连接到HSS，执行订阅操作和维护用户配置。在这种设计方式下，P-CSCF能被认为是一个SIP边界代理，它是SIP请求到达的第一个处理点，然后进一步将请求路由到目的地（DNS 查询）。I-CSCF和S-CSCF代表了网络服务器，支持用户注册和服务访问。在建立会话和会话期间，S-CSCF作为一个SIP代理服务器接续终端间的消息，因此它也可以看做是一个应用服务器。

应该注意CSCF模块不被作为真正的SIP代理，而是被作为所谓的背靠背用户代理，能终止一端的呼叫，并在另一端发起同样的呼叫，而不是简单的接续。所以在IMS环境中SIP的端到端信令模型不再有效。确切地说，它是一个封闭的网络环境，因为一些信令消息或者SIP消息头（SIP规范所允许的）都终止于P-CSCF，而不再进行转发。这就强制了它必须以网络为中心来进行服务构建，而不是用户或者第三方提供者。

IMS最初是为移动网络而设计，接下来又经过3GPP标准化的。TISPAN（Telecoms & Internet converged Services & Protocols for Advanced Networks，电信和互联网融合服务及高级网络协议）作为ETSI关注于下一代网络的标准化小组，最近将IMS纳入了固定网络架构。利用这种方式，IMS可以支持固定网和移动网，同时也迎合了固定网和移动网融合的趋势。

3.1.5 开放API

Parlay API的出现首次将运营商的网络和服务平台向所谓的第三方提供者开放，使其能有机会提供服务。Parlay API是由运营商和制造商驱动的Parlay联盟所提供的标准化应用程序接口，它使第三方应用能够通过一系列开放的、标准化的接口利用网络功能。对于移动网络通信系统，3GPP已经采纳Parlay规范作为OSA，同时对其进行发展延伸。

图3-13描述了OSA的结构并指出了OSA网关的API。此API允许应用服务器上的应用程序去访问服务能力服务器，该服务器提供了具体的服务和网络控制功能。服务能力服务器将OSA API中的可用方法映射到特定的网络协议，因此可以对应用程序屏蔽掉特定的网络细节。表3-1提供了最重要的服务能力功能的概述，这些功能已经由OSA标准化了。

表3-1 OSA服务能力功能

能力服务器	描　　述
呼叫控制	建立、控制基本呼叫和多媒体会议
用户交互	获取终端用户信息，发布公告，发送短信息
用户定位/用户状态	获取用户位置和状态信息

（续）

能力服务器	描　述
终端能力	获取用户终端能力
数据会话控制	对数据会话进行控制
通用消息	进入邮箱
连接管理	提供 QoS 保证
账户管理	获取终端用户账户
基于内容收费	根据终端用户使用的应用/数据计费

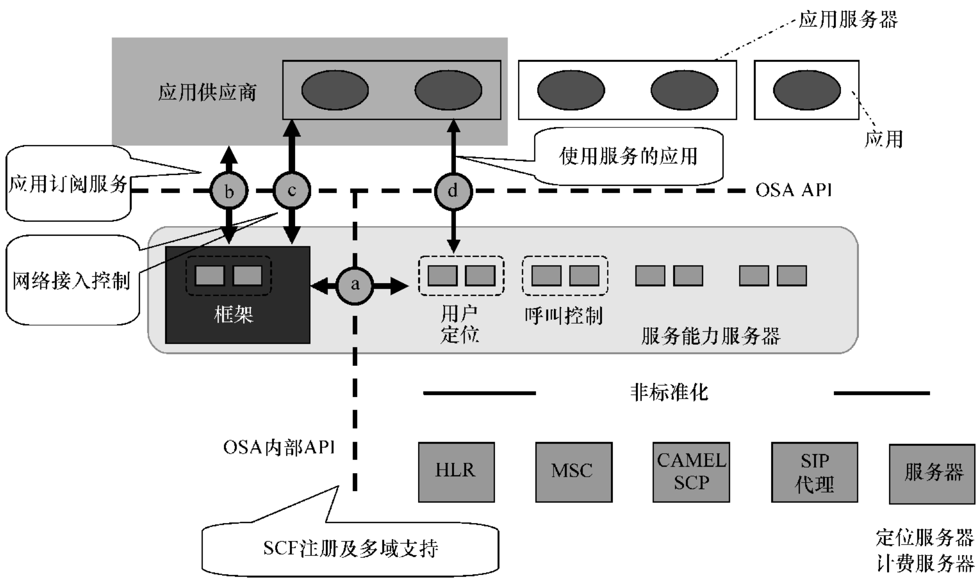


图 3-13　OSA 体系架构

除了标准化允许访问服务能力功能的接口，OSA 进一步标准化了两个接口类。Framework 接口负责控制应用程序在使用服务能力功能之前访问 OSA 网关（见图 3-13b 和 c）。内部的 OSA API 描述了服务能力功能的注册和应用程序接入的鉴权。

然而，OSA API 仅仅是向第三方服务提供者开放移动通信系统的第一步，它的使用仍然是复杂的，且比在互联网上创建服务需要更多电信网方面的知识。

3.1.6　移动互联网

互联网及其应用如万维网的成功是因为其易用性。用户通过浏览器可以轻松访问互联网，不仅仅是通信服务提供者，任何类型的企业或私人用户都可以轻松

地创建新应用。很明显，运营商也想要在移动通信网络上借鉴互联网的成功。为了提供像万维网的信息服务给移动用户，出现了不同的服务平台。在这一部分，我们将讨论无线应用协议（WAP）和日本的i-mode系统。虽然拥有相同的目标，但是它们的成功却是不同的。WAP最初关注于优化通用平台基础设施以适应低数据传输速率的无线传输，最终形成了一个新的协议架构，而i-mode关注于应用程序，随着新的商业模型的引入，使第三方服务提供商们能够容易地为移动用户提供新的应用。

1. 无线应用协议

著名的WAP[20]移动互联网服务平台是为移动用户设计的，提供基于IP的、类似万维网的信息服务。其主要的注意力放在基础架构上，因此使得无线连接的互联网协议栈得到整体的优化，并提出了一种新的、不同于HTML的优化标记语言——WML（Wireless Markup Language）以适应移动终端有限的显示能力。图3-14展示了WAP协议栈与用于Web浏览的互联网协议的对比。工业组织开放移动联盟OMA（Open Mobile Alliance）已经标准化了WAP。

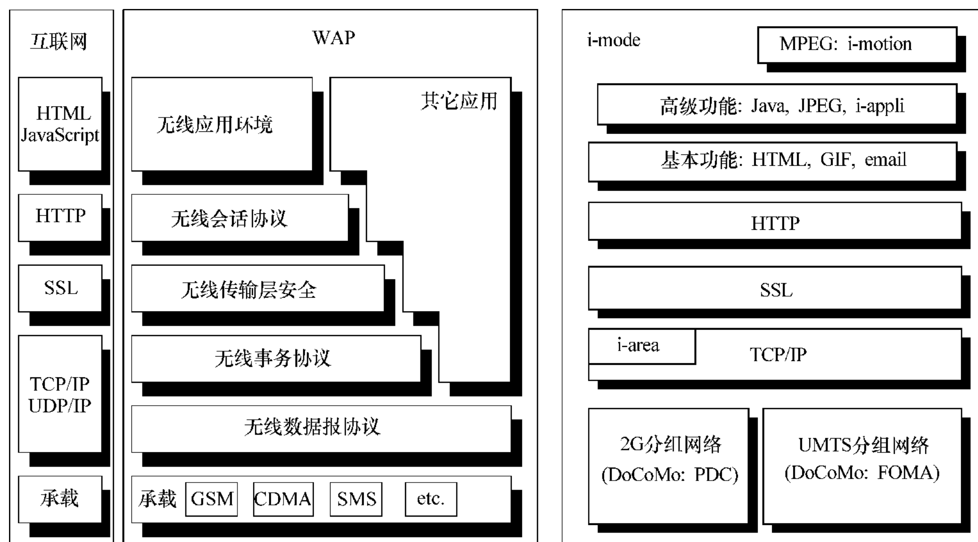


图3-14 WAP和i-mode协议栈比较

图3-15给出了WAP服务架构的概述。WAP架构基本上由WAP客户端、WAP代理及WML服务器组成。在移动通信网络中，WML通过WAP协议栈传送到WAP服务器，取代了互联网中HTML通过HTTP/TCP/IP协议栈的传输方式。WAP既支持电路交换承载（如GSM的数据通道），也支持基于分组方式的承载（如GPRS）。WAP代理作为接入WML服务器的网关，WML服务器可以部署在

移动网络运营商的分组域中，也可以通过互联网直接接入。此外，WAP 还提供了一个为移动通信设备开发应用的环境（Wireless Telephony Application，WTA）。

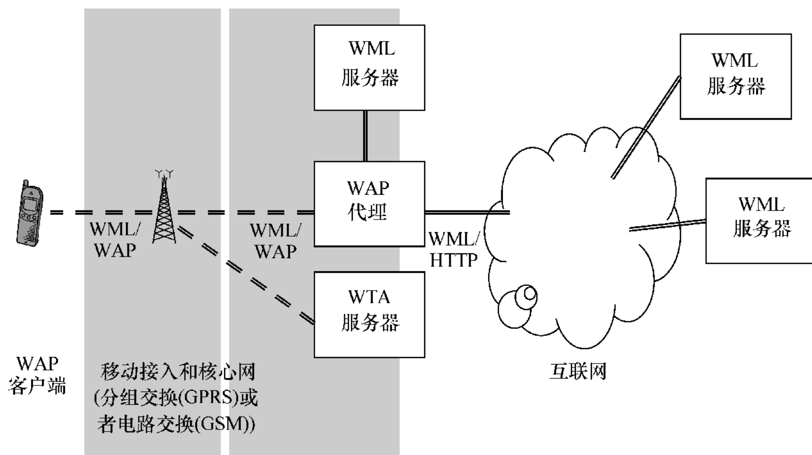


图 3-15　WAP 服务体系架构

2. NTT DoCoMo 公司的 i-mode 服务

i-mode 发展的目标和 WAP 是一样的：为移动用户提供和互联网一样的信息服务。但是从一开始，它将重点放在了服务和应用的提供上，而不是研发出一个理想的传输技术。i-mode 的成功证明了这一决策的正确性。现今，日本大约有 4600 万的 i-mode 用户（50% 的日本手机用户和 91% NTT DoCoMo 的客户）。另外，i-mode 的概念和技术也在世界围内得到推广。目前已经有 15 个国家的服务运营商部署了 i-mode 的服务（来自 NTT DoCoMo，2006 年 6 月）。

i-mode 的服务架构（见图 3-16）和 WAP 很相似，显示出一些简化的协议和附加功能，如电子邮件所做的那样。i-mode 的客户端通过一个基于分组的网络连接到 i-mode 服务器，服务器上有 i-mode 的网站，这里提供了所有官方的 i-mode 内容提供商，用户可以订购他们的服务或者通过 i-mode 网关连接到互联网以获取信息和使用电子邮件功能。另外要说的是 i-mode 官方服务器必须经运营商认证，非官方的服务器可以有无限多个，通过 HTTP 用于提供与 i-mode 一致的页面。

同没有取得较大商业价值的 WAP 相比，i-mode 能在商业上取得成功主要有以下几个因素：在传输层，i-mode 使用了分组交换网络，和 GPRS 很相似，在用户侧都利用无线链路的。我们介绍 i-mode 就势必会介绍 NTT DoCoMo 的分组网络。WAP 可以承载在多种传输层上，包括 GSM 数据通道和 SMS，这对 i-mode 服务提供商来说这意味着一种新的威胁。但是正因为这个原因，i-mode 服务从

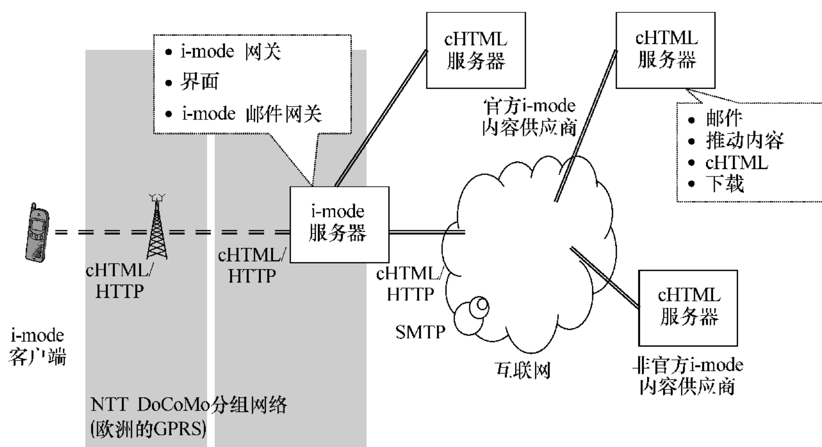


图3-16　i-mode服务体系架构

一开始就收到了更好的客户接受度，因为有了更高的数据传输率和在线率，如基于分组计费。i-mode与WAP相比，更接近互联网标准。WML经过高度优化，但是并不难学。i-mode页面由cHTML语言编写，它是HTML的一个子集，为了提高性能而删除了某些功能。另外，i-mode使用标准的HTTP/TCP/IP在无线链路上传输。通过这种方式，协议的优化并没有使性能得到明显的提高，相反i-mode的性能是通过内容来控制的。对于认证i-mode网页，每页的布局和大小都是由运营商提供，并且严格控制的，这样可以减少下载时间从而提高客户的接受度。

一直到现在我们讨论的都是技术因素，成功的另外一个主要因素在于商业模式的革新（见图3-17），该模式被认为实现了一种新的移动服务提供方式，它在接受第三方内容提供商这个角色的同时，也成就了移动运营商和内容提供商双赢的局面。经过认证的i-mode内容提供商提供他们的内容到运营商的网站，供运营商用户使用并且不收取服务费用。这个服务费通过运营商收取，并将一部分支付给内容提供商，运营商只留下很少的一部分（在日本为9%）。运营商主要收益来源是流量计费和客户绑定，因此他们会鼓励服务提供商提供更多的创新服务，同时他们会依赖运营商进行管理和收费等。运营商不仅仅控制i-mode在技术方面的兼容性，也控制i-mode认证网站的数量和类型。新的内容提供商必须提供与现有服务不同的服务。这样用户可以得到内容更丰富和平衡的服务，而不是无奈地接受过多类似的服务。

在与移动手机制造商的关系上，i-mode也与WAP是不同的。WAP标准需要应付许多不同的设备配置，实际上一个WAP页面几乎需要在所有主流手机的浏览器上测试，以保证其能正常显示。NTT DoCoMo与手机制造商的关系在欧洲更

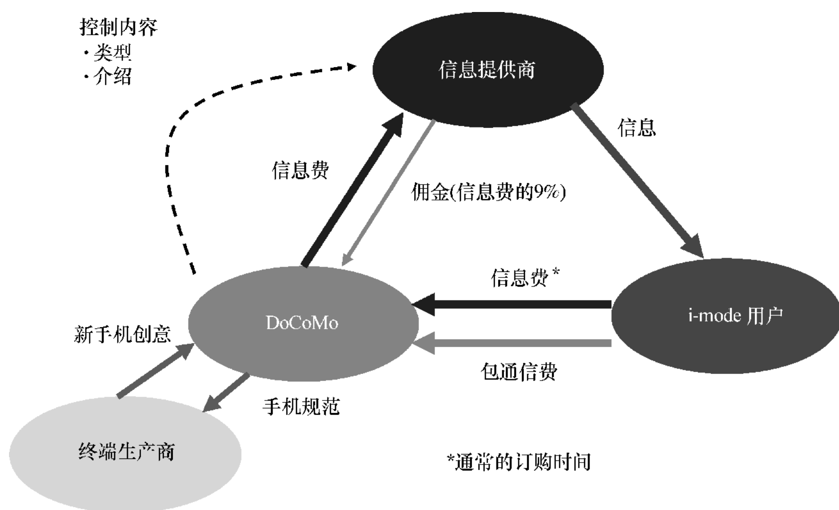


图 3-17　i-mode 商业模式

为密切，比如，他们会与设备商一块准确详细地规定设备的能力，并加上独一无二的商标，这从技术上保证了设备的兼容性。该品牌在欧洲至今仍可见。

现在 i-mode 已经在日本以外地区可用，其商业模式也被其他运营商所采纳。由于在移动手持设备中引入了 XHTML（这些手持设备原支持 cHTML 和 WML），WAP 和 i-mode 正在逐步地融合。并且 WAP2.0 使用 HTTP 和 SSL 代替 WSP、WDP 和 WTLS，从而实现更为平滑的互联网整合[20]。

总而言之，从移动互联网中的 WAP 和 i-mode 我们可以看出，一种新的服务要想在移动通信领域取得成功，仅仅有技术的支撑是不够的，与内容提供商的联合，特别是用户的接受度也应该深入考虑。

3.1.7　移动服务平台需求

基于以上讨论，我们可以总结出服务平台的主要需求（不仅仅是移动服务平台）：

1）为服务和应用开发者提供接口，可以从网络细节和支持的增值功能中抽象出来，如用户支持、接入、移动性或定位功能。

2）服务的性能和可靠性需与用户在固定网中的体验相同。

3）可直接接入互联网并使用其中的服务，或者使用互联网设施构建互联网兼容的分组服务。

4）向外部服务和内容提供商提供接口，以增强其服务和应用能力。

我们已经了解了服务平台演进过程，服务平台由系统内部隐式包含的部分演变成一个独立的显示存在的组建，这成为移动通信系统取得成功的关键。并且，

服务平台的研究不会停滞不前，它会如系统其他部分一样继续演进，为未来移动系统提供服务。我们将在下一节介绍下一代服务体系架构的功能实体以及其所面临的挑战。

3.2 下一代服务体系架构

移动通信向用户提供越来越丰富多彩的服务，并发展成为人们生活中必不可少的一部分。如3.1节所述，蜂窝技术因受到互联网技术的影响而得以改进，并希望像互联网服务一样取得巨大成功。目前泛在计算的研究趋势愈演愈烈，它引入了在不同设备环境中进行交互的概念，如传感器和驱动器。并且，为了能够支持宽带多媒体服务，业界发布了一系列新的标准来支持高数据传输率的移动多媒体应用。

在这样一个解决方案日趋多样化的时代，下一代移动通信系统的成功就取决于其所能提供的服务和应用。未来的服务平台应该能集成各种提供开放接口给服务和应用提供商的模式。一些新的软件技术，比如说Web服务和语义Web将会有可能在未来的服务提供中发挥极其重要的作用。一些新的添加了在3.1.7小节提出的附加需求的解决方案也不断涌现。例如，当把服务和应用按照用户的实际需求或偏好适当地“裁剪或变换”，用户的接受程度就会大大的增加。另外一个例子是P2P服务，在这种服务中移动用户可以直接跟其他用户进行交互而不需要一个集中的控制。一个设计良好的下一代服务平台应该可以提供各种各样的方法和途径以允许提供商创建各种富有创意的服务，并不断地满足用户的需求。例如，第三方接口可以形成一条服务提供者生态链。此外，语义技术或许能够帮助把用户环境相关的上下文知识结构化。

3.2.1 挑战

基于上述考虑，我们将从不同角度阐述下一代系统所面临的挑战。从服务生命周期的角度来看，我们将更多地基于用户角度的考虑去预测扩展服务的生命周期；从网络的角度来看，新出现的通信系统（包括扩展的蜂窝系统或者无线LAN、无线传感器网络等）都可被考虑用于服务的提供，这被称作“泛在通信”。系统的开放性将导致新的用户角色模型以及挑战的出现，所以当某个系统尝试寻找成功的商业模型时，我们将不得不考虑系统的开放性。

众所周知，服务生命周期由服务生成、服务部署、服务使用以及服务终结组成。从用户的角度来说，我们应该更关注服务使用，因为这是服务生命周期中惟一能够被用户感知到的部分。这里将描述一个基于用户角度的扩展，它包括服务选择，服务执行以及服务终结模式。图3-18展示了一个扩展的服务使用的生命

周期。基于用户当前的上下文信息，服务能够被用户选择。因此，信息处理是在服务选择阶段之前的。伴随着系统向更多的服务提供者开放，每一类型的服务对用户来说不再是一个服务，而是大量服务，这使得用户不得不从中选择。未来的服务提供系统应当能够帮助用户去发现可用的服务，从而使用户能够做出选择。另外，服务也能够实时地由其他服务生成。为了适应用户的使用喜好和上下文背景，服务在执行之前应经过调整和配置。持续性的调整在服务执行期间也是必要的，直到服务终结。如图 3-18 所示，虽然各阶段的顺序不是固定的，但图中所展示的服务使用生命周期可作为对服务生命周期的一种扩展。

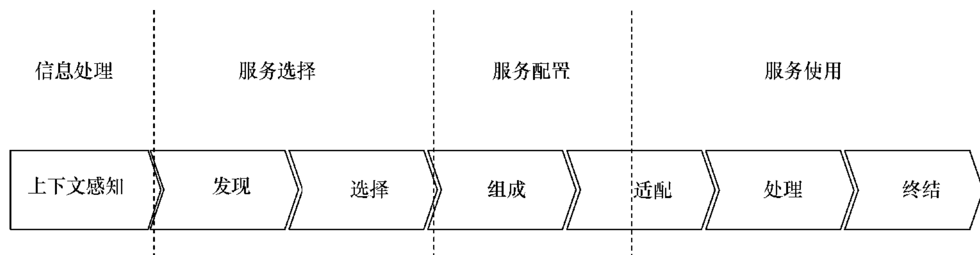


图 3-18　服务使用生命周期的扩展

从通信的角度来看，未来移动系统[22]中的服务平台必将向无线数据网络、传感网络这样的泛在环境扩展。这里我们给出一个泛在服务的广泛定义，泛在服务指的是非现有移动系统通信环境中的服务。那么泛在服务就包括利用环境信息（例如由传感器网络收集的上下文信息）的移动服务，能在异构环境中接入而且可用的移动服务，能使用非蜂窝式网络架构的移动服务（例如 P2P 服务）。图 3-19展示了对传统服务平台的扩展。移动运营商的服务平台可以通过与传感系统和 P2P 系统的结合与交互进行扩展。可以考虑将移动电话作为网关，在桥接不同网络时发挥重要作用。新兴网络环境不仅可以将服务传递到设备，而且还能提供自身的有用信息，利用这些信息，服务平台或者第三方服务提供者可以更好地定制服务。

如 3-19 左图所示，移动服务提供系统的角色由于市场竞争者的参与（例如内容提供商和无线接入数据网络提供商）已经变得很模糊。由于数字移动性和互联网网关选择，传统用户绑定机制正逐渐消失，运营商正试图探寻一种新的稳固用户和增强用户关系的方法。一种方法是通过提供个性化服务增加用户关注焦点，而另一种方法是第三方提供商充分利用运营商已有的丰富的客户资源来提供用户定制服务。这些第三方提供商所利用的服务个性化特征包括简单的服务接入和运营商作为值得信任的角色进行计费和服务交付。

基于以上在服务提供、网络和角色模型等方面的考虑，移动服务平台新的需

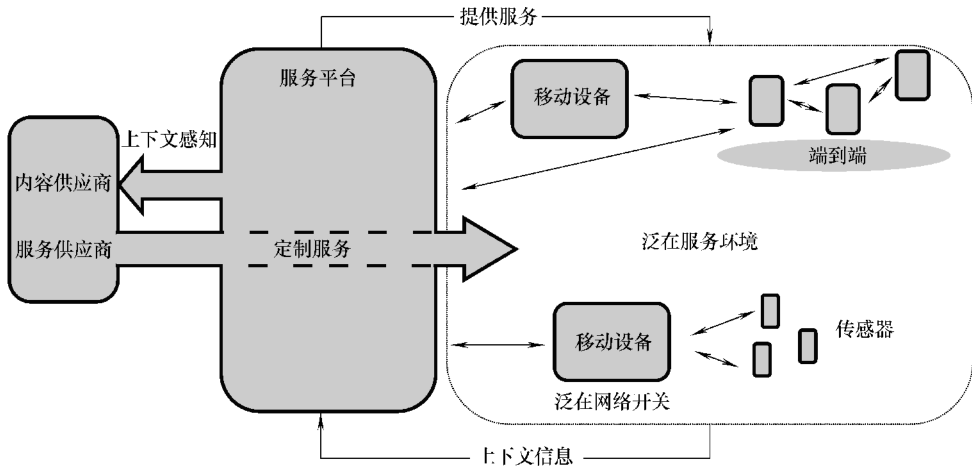


图3-19 服务平台向泛在环境的扩展

求如下所示：

1）基于个人用户和用户群（个性化、社区）；

2）考虑应用可执行情况和所处环境（上下文感知）；

3）考虑用户周围所有可使用的设备并支持使用这些设备；

4）考虑现有的通信技术（感应网络。P2P通信）；

5）允许异构环境以及不同运营商网络切换时的无缝服务；

6）支持更为多样和丰富的服务和应用（移动多媒体）；

7）服务接入、使用和计费的简单性；

8）集成了运营商、服务提供商和第三方服务组件提供商的商业模式；

9）通过信任和声誉的用户绑定。

实现了以上需求的服务平台技术起源于计算机科学、电子工程以及其他学科。我们在以后章节将会看到，不同领域的技术整合可提供创新型的解决方案。

3.2.2 通用服务特征

本节将主要列举新兴服务平台最重要的服务特征。一些主要内容将会在第五章展开详细叙述，其中涉及移动中间件和包括支持上述新兴性能的移动系统的中间件功能描述。这些中间件功能可以看成统一服务特征的更高层或是用于服务支持的中间件组件，这些内容将在以下部分列举说明。

除了传统的服务支持的中间件特征，比如数据仓库、性能管理和事务管控，通用服务特征还包括：

1）发现和广告功能，可动态发布和查询不同域的服务信息。

2）支持内容适配。

3）群组成员管理。

4）允许用户与服务提供商协商协定。

5）合法的合同服务。

6）域计划模式为概念模式提供可处理的、上下文信息使用的数据结构。

7）动态地用其他服务和服务组件组合服务，也包括其它域的服务。

3.2.3　泛在服务特征

根据 3.2.1 节中泛在服务的定义，其有以下多个特点：

1）使用环境信息（例如从传感器获得的上下文信息）。

2）在异构环境中可接入而且可用（任何时间、任意地点）。

3）依赖于非网元结构网络（比如 Ad Hoc 点对点服务）。

服务平台需要增强其功能才能支持泛在服务的提供。在泛在环境中必须有网关功能实体，它用于搜索和获取信息。而且泛在设备不仅能够感知到服务，还应该能够激活泛在服务。3.2.2 节在介绍无缝接入特征时已经讨论了无缝接入和使用。然而在这里，我们强调无缝泛在服务应当能够感知用户的周围环境，从而减少与用户之间交互。在 3.3 节中，我们将描述一个能够实现移动性无缝接入的服务平台组件。当提到泛在服务时，我们不得不考虑这样的服务平台，它不基于任何服务器组件，只是利用消费者设备来提供服务。然而这从传统运营商看来是一个巨大威胁，不过从服务提供商的角度来看，它也提供不少好处，比如节省设施、拓宽用户定义服务的范围。我们将在第 4 章介绍 P2P 服务平台，并且会重点介绍移动 P2P 服务平台。

3.3　泛在服务示例：会话移动性

未来的移动通信中，用户很可能有多种通信设备（移动电话、IP 电话、可穿戴设备等），它们的接入网络都不相同（如无线局域网（WLAN），固定 DSL 网络以及移动通信网络）。这样的通信环境通常被称为泛在环境，如图 3-20 所示。

本节我们将会以泛在服务的一个具体系统为例，来说明它是如何在异构多设备环境中实现的。我们会特别重点介绍由应用层支持的关于移动性方面的无缝服务接入以及服务连续性问题。此外，我们会详细介绍一个基于 IP 的服务平台系统的实现，它使用多媒体信令控制中的 SIP 协议。

在介绍系统之前，我们需要首先了解移动性的分类。我们众所周知的终端移动性只需要网络层支持切换和漫游即可实现，而在以用户和服务为中心的网络中，其他类型移动性也日益受到人们的重视。据最新资料显示，应用层除了支持

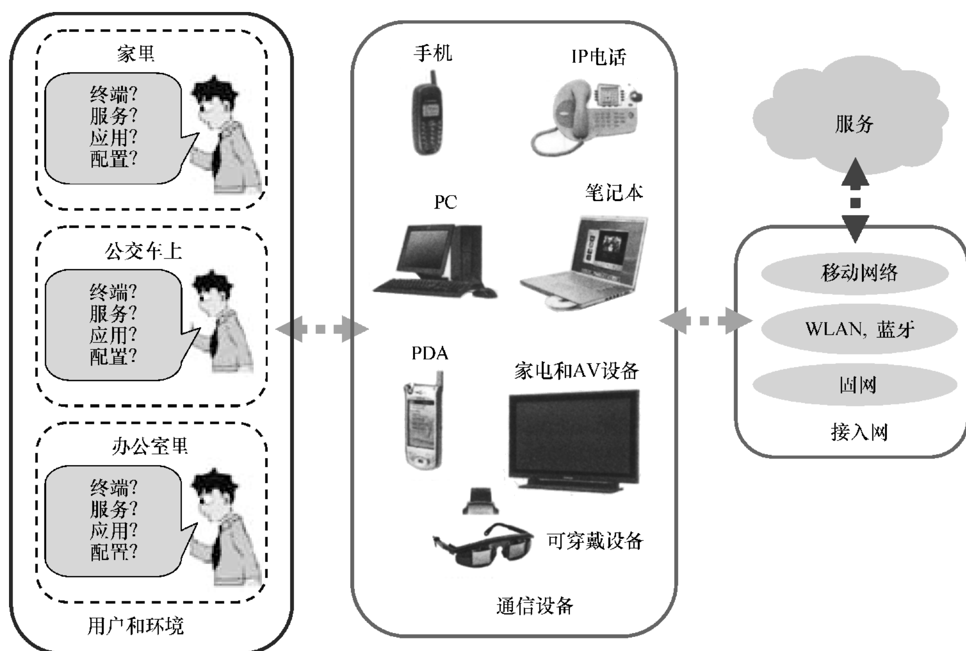


图 3-20 泛在和异构环境

终端移动性外还可以支持如 SIP[24]、ICEBERG[25]、TAPAS[26]、SSE[27]和 BSPM[28]等移动性。利用应用层信令协议来进行移动性控制比较有优势，例如在 SIP 中，使用 SIP URI（全球资源标识）来惟一地标识用户，独立于网络（IP）地址。在移动通信中，网络层移动性和服务层移动性仍需要更好的合作来给用户最大程度的自由。

3.3.1 移动性分类

最初只有两种移动性，分别为终端和个人移动性。随着用户需求的不断增长，出现了另外三种移动性：服务移动性、配置移动性和会话移动性。

1. 终端移动性

终端移动性指的是终端在改变位置时，仍能维持当前的通信（一个终端—多个网络）。终端移动可以分为两类：水平切换和垂直切换。水平切换指的是终端在同一接入网的不同小区之间移动；而垂直切换则指终端在不同接入网之间移动，如从 UMTS 切换到 WLAN。如图 3-21 表示了终端与网络之间的动态连接关系。

2. 个人移动性

个人移动性指的是终端用户可以在任意位置的任意终端上发起、接收呼叫和使用电信服务，用户移动过程中，网络仍然能标识该用户（一串数字/地址—多

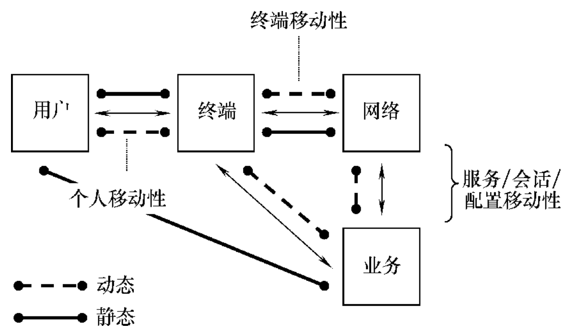


图 3-21　移动性分类

个终端）。个人移动性是基于用户唯一身份标识的（如一串私人数字或者一个私人地址），用户与终端之间的映射应该是动态的。

3. 服务移动性

最初，服务移动性的概念来源于虚拟归属环境（VHE），它允许用户使用和维持与用户位置和终端无关的服务，例如可以使用与终端类型和网络运营商独立的个人地址簿。一般来说，服务移动性指的是服务环境跨网络和终端的可移植性，具体来讲，就是指外观、功能和用户界面等（一个服务—多个终端和域）。如图 3-21 所示，要保持终端、网络与服务之间的动态关系，必须解决以下两个问题：首先，如何移动服务；其次，如何发现服务和设备。举个例子，用户移动到一个新环境中，他（她）或许会想利用新环境中功能更强大的设备来享受服务（设备发现）。

4. 配置移动性

配置移动性是服务移动性的子问题。配置移动性指的是某一个服务环境中的个性化信息和功能跨终端、网络（甚至是服务环境）的可移植。这就意味着配置移动性可以在任意设备、任意位置获取用户配置和用户偏好。

当前的解决方法是各种服务都可以使用配置文件，数据配置文件通常存储在本地，如用户的终端（GSM SIM 卡、PDA 或者个人计算机）。然后这些配置通过手工在各服务之间同步。还有另外一种解决方法是，一个数据配置文件只针对一个具体的服务，并存储在本地或者中心服务器上，但这种解决方法通常是私有的或者只能在特定的网络中。目前并没有一个针对不同网络、服务和终端的通用解决方法，所以就导致了现在用户拥有多个具有相同信息的个人数据库。例如用户在很多地方存储通讯录，如移动电话、SIM 卡、办公室电话、PDA 和网页浏览器地址簿中。

5. 会话移动性

会话移动性指的是用户在保持当前会话不中断的情况下，将会话切换到另一

个终端。它也是服务移动性的一个子问题，描述已运行服务的转移（或许包括当前服务环境）。举个例子，用户在回家的路上进行视频聊天，当他回到家后，他希望将通话切换到能力更强的终端上，比如更大的屏幕或者音质更好的传声器。

3.3.2 支持SIP的泛在服务环境中的服务移动性

为了说明基于IP环境的泛在服务所支持的功能及其实现，接下来我们将介绍基于SIP的体系架构，它可以为服务提供基于上下文的转移。当用户拿着移动设备进入新的区域（如公司的会议室）时，它会：

1）在用户环境中动态和自动发现所有可用的异构设备（可编程的和不可编程的）及其处理能力（例如支持的媒体类型和编码方式）。

2）转移用户配置到发现设备的配置中。

3）当在一个设备上改变用户配置时，更新所有其他使用中的设备配置。

4）将正在进行的实时多媒体会话（视频呼叫）转移到当前环境中新发现的一个或多个设备上。这个过程也包含了会话的分离能力，例如将音视频会话分离为两个子会话（一个音频，一个视频），再将这两个子会话分别转移到不同的设备上。

5）将刚才转移的会话重新转移到移动设备上。

此外，会话当事人对会话的转移应该是不感知的，而且用户的设备也不需要进行SIP客户端软件的更新。所有功能的实现都应该使得用户的输入最少。

接下来的示例进一步说明该功能。

1）配置移动性（1）。机场公共电话亭包含一个电话和一个单独的ID卡阅读器。旅行者Tony进入电话亭，插入个人ID卡，里面包含有他的SIP信息和鉴权信息。电话亭中的令牌阅读器注册Tony信息，将其SIP地址作为他的联系地址。亭中的电话也已自动更新为Tony的注册信息。现在Tony可以利用归属服务器作为输出代理，而此时电话里已有他的电话薄和其他个人信息。当Tony再次插入ID卡时，阅读器则会将电话亭中公共电话上的用户配置文件删除。

2）配置移动性（2）。Charlie正在会议室中使用PC软终端电话加入一些联系人地址。他的每一次输入都会更新其漫游配置，并会自动给PDA发送通知，此刻正在进行设备同步。他带着PDA离开房间，但忘记将软终端上的配置信息删除。此时Charlie进入到另一个房间，他的PDA在收到一个从蓝牙设备发来的协调消息之后，则会在另一个房间更新他的位置，软终端电话也识别出他已经不在房间中，从而删除他的数据不让其他用户看见。由于进行了同步，他的PC也

会更新他电话薄中的改变。

3）会话移动性。Alice 到位于巴黎的公司下面的一个子公司出差，当她还在去的路上时，接到同事发来的视频会议请求，她用自己的手机接受了请求。她到达公司后，就立刻走进一间配有多种通信设备的办公室，这时她的手机也就发现了屋子里的通信设备。为了享受更好的音视频效果，她将视频流转移到房间内的大屏幕上。而由于该大屏幕没有音频设备，于是她将音频流转移到办公桌上的 IP 电话，并利用它的扬声器和传声器，这样就可以腾出手来干别的事情了。

基于以上需求，我们的架构由 3 个基本元素构成：用于设备（D1，D2）之间转移用户配置和会话的信令流（实线）；服务器（代理）和设备管理器（DC，DM）；为用户屏蔽转移时分流的多设备系统（DM）和本地设备发现（目录代理，虚线所示），见图 3-22。

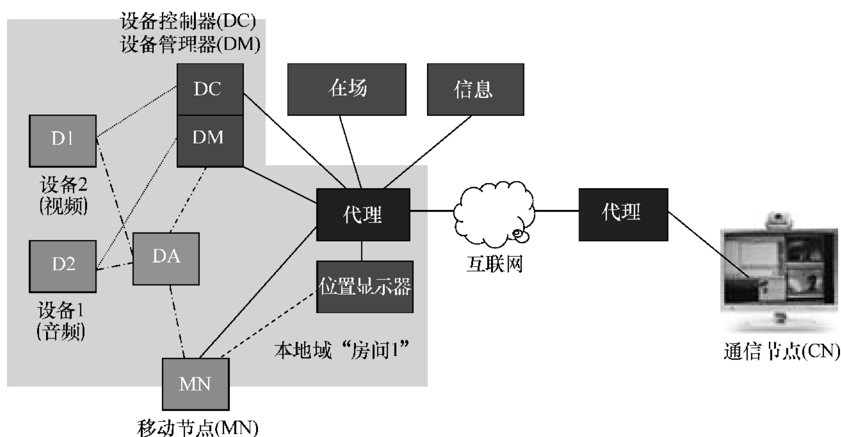


图 3-22　会话移动性系统概述

在许多研究项目中都把服务移动性作为泛在服务广泛讨论的专题，目前已经有好几种解决服务移动性问题的方法。但是，大部分现有方法要么是基于私有协议实现[26]，要么需要终端进行软件升级。于是引入新的组件，例如在会话转移之前就插入媒体流的媒体代理[30]。而其他的解决方案都集中于某一具体服务类型，如流媒体[27]。再看看移动通信系统中 IP 服务平台所使用的协议，如 IMS 中使用的 SIP，对于信令服务和配置转移它已经有了基本机制。

接下来介绍的解决方法就是基于标准 SIP，使用标准的包结构，如 SIP 事件结构。不需要对 SIP 进行非标准的扩展，并且被叫用户不能感知会话的转移，这是由 SIP 自动完成的，除非需要改变媒体流终端。

3.3.3 实现会话移动性

会话移动性指的是当前的会话在设备之间进行转移，可能还包括将会话流分离到多个设备中。在移动环境中，会话转移发生在移动节点（Mobile Node，MN）和相关节点（Correspondent Node，CN）之间，指的是将当前正在进行的会话从MN转移到本地环境中的设备，当用户离开时还能再转移回MN。

会话转移可分为两种模式：移动节点控制和会话切换。在移动节点控制模式下，MN始终控制着会话信令，即使会话在本地设备和CN之间传输。在会话切换模式下，当会话转移到本地设备后，MN放弃对当前会话的控制。CN与本地设备建立起新的会话，并将代替MN与CN当前正在进行的会话。

每种传输模式都有两个选项，转移整个会话到同一设备或者将会话分成多个流转移到多个设备中。整体会话转移是比较简单的情况，它将整个多媒体会话从一个设备转移到另一个设备，例如，一个语音呼叫可以转移到一个IP电话中，而一个语音视频呼叫可以转移到一个视频电话中。如果本地设备能支持在用户MN上不支持的媒体类型，那么他仍可以移动会话并且添加媒体类型。这种切换适用于当用户到了他的办公室，并且决定将当前的音频呼叫转移到房间内的视频电话，以得到更好的享受。如果用户将正在进行的音视频会话转移到只支持语音的IP电话，那么媒体流的数量将会减少。

在分离会话转移中，一个多媒体会话将被转移到多个设备中，这有可能包含移动节点，例如用户将一种媒体类型转移到另一设备中，同时通过邀请其他设备来保持本地会话或者为会话添加媒体类型。当然，会话中的所有媒体都有可能会被转移到其它设备中。将单个媒体流进行划分，会话可以得到进一步的分离。例如一个手机或者PDA用户，在进行视频会话时，会通过设备的照相机将自己的视频传输出去，而在观看其他参加者时会觉得图形太小，因此他可能会把视频通过投影仪放映出来，而自己的头像仍通过设备上的照相机传输出去。

如前面所介绍，从体系架构中可以看出，SIP在将当前会话从MN转移到本地设备或从本地设备转移到MN时起着信令控制的作用[31,32]。

1. 移动节点控制模式中的会话转移

我们这里举的例子使用了SIP第三方呼叫控制（Third Party Call Control，3PCC）[33]。在这种模式下，MN作为控制器，负责与本地设备建立单独的会话并更新与CN之间的会话，这样本地设备和CN之间就能建立起媒体连接了。MN发送INVITE请求到本地设备建立一个新的会话。本地设备回复响应，并在SDP中包含自身的媒体参数，该参数也会用于向CN发送的RE-INVITE请求中。一旦CN与本地设备都有了对方的媒体参数，它们之间就可以直接建立媒体会话了。如果要将会话分离到多个设备中，MN会对本地每个设备都分别发送INVITE请求建

立新的会话。在 CN 侧，则会根据收到的 SDP 消息对正在进行的会话更新。

2. 会话切换模式中的会话转移

如架构中所描述，会话切换模式使用 SIP REFER 方法[34]来发送请求，在本地设备和 CN 之间建立新的会话，代替当前会话。MN 发送 REFER 给“仲裁者”，请求它与一个具体的目标 URI 联系，“仲裁者”可以是本地设备也可以是 CN。根据架构所示，我们将 REFER 请求发送到本地设备。本地设备收到请求后，向 CN 发送 INVITE 请求来初始化一个新的会话。用“Referred-By”头来标识 MN，他是一个新会话的发起者。用“Replaces”头来请求一个新会话，代替当前 CN 与 MN 之间的会话。

接下来将介绍 3 种方式进行分离会话转移，它们都使用 REFER 方法。每一种方式，我们都假定会话已经在移动节点（MN）和协调节点（CN）之间建立，而且 MN 在当前位置发现了本地设备（一个音频设备，一个视频设备），并利用它们来继续当前的视频电话会议。

第一种方案中，MN 发送 REFER 消息到 CN，要求 CN 给本地设备发送 INVITE 消息，加入到会话中。当本地设备都接受了来自 CN 的呼叫请求之后，CN 会通知 MN 此次新会话的建立，但仍通过使用 NOTIFY 方法代表 MN。最后，MN 与 CN 之间的会话终止。这种方法的一个缺点在于，进行会话转移时，呈现给 CN 的是两个新的呼叫，而不只是在会话另一端两个设备之间的转移。

第二种方案是 MN 发送 REFER 消息给每一个本地设备，要求它们给 CN 发送 INVITE 消息。REFER 消息中包括一些消息头（如 Referred-By，Replace）、关于当前会话的必要信息（dialog，call-id，tags 等）和 MN 鉴权信息。此鉴权信息用于对本地设备进行鉴权，这样 CN 就会自动接收来自本地设备的 INVITE 请求。由于 CN 仅仅需要更新一个新的媒体连接，而不需要建立新呼叫，所以这种方式较之第一种可以更平滑地进行会话转移。然而由于在 CN 看来，呼叫转移仍然是两个不同的呼叫，CN 仍需各自结束与本地设备之间的会话。

第三种方案是将 REFER 发送到一个专门的功能实体—设备管理器（Device Manager，DM）。DM 用来创建“多设备系统”，即一个系统连接多个设备。DM 中有一个 SIP 背靠背用户代理（B2BUA），它既可以做 SIP UA，也可以做 SIP 代理服务器。这种方案中，MN 只需要将 REFER 消息发送到 DM，而由 DM 分别给本地视频和音频设备发送 INVITE 请求。本地设备则会回复一个 200 OK 消息到 DM，其中包含了 SDP 消息，里面包含了倾向的媒体连接（本机地址）和媒体参数（媒体名称和传输端口）。然后 DM 向 CN 发送 INVITE 请求，其中包含了音频设备和视频设备的媒体信息。类似之前的方案，INVITE 消息中还包含了建立新会话的必要信息和消息头，用以代替 MN 与 CN 之间的当前会话。图 3-23 描述了这种方案。

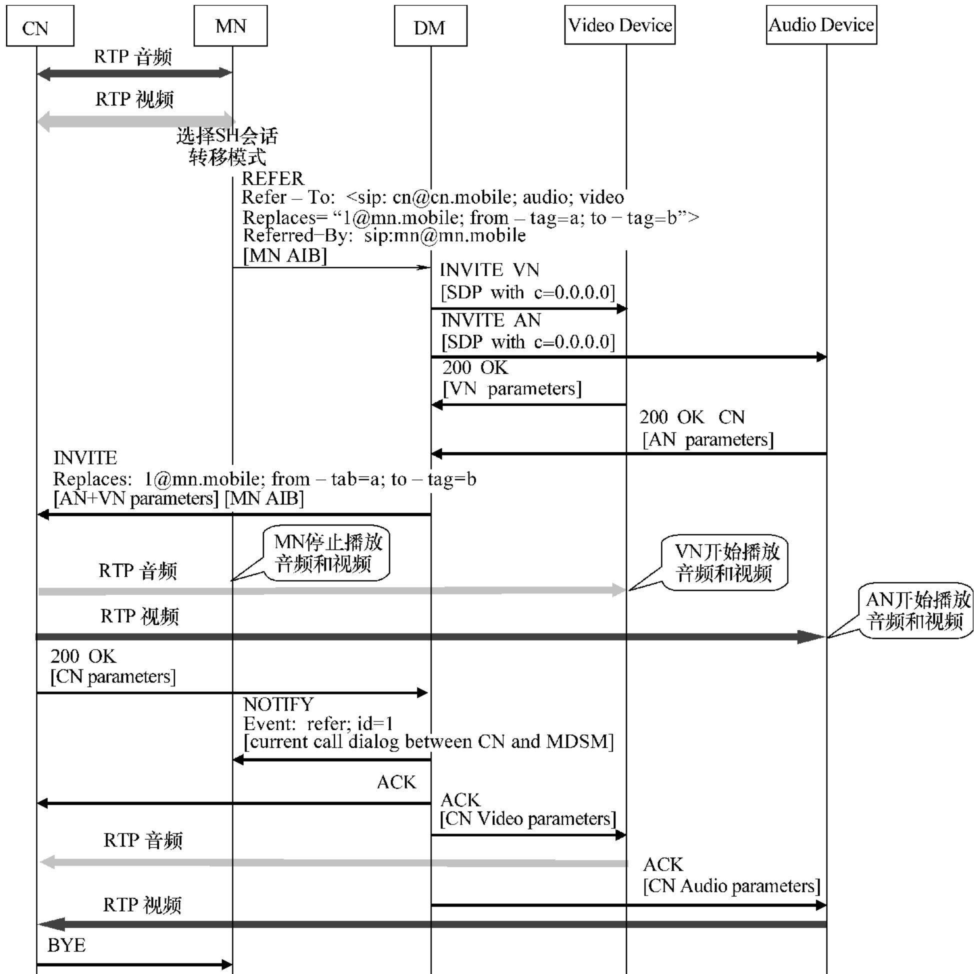


图 3-23　会话切换模式下将会话流分离到两个设备的会话转移流程

比较以上 3 种分离会话转移方案，从 CN 侧来看的话，使用 DM 这种方式似乎是最平滑也是最简单的。由于 CN 直接与 DM 进行会话，所以无需单独与每一个本地设备终止会话。

3.3.4　发现泛在设备

由于各组织领域可能会很大，而且设备也会频繁的添加和删除，因此，动态的、去分布式地维护各设备精确的位置信息将非常重要。

如架构图所示，服务定位协议（Service Location Protocol，SLP）[35] 用来在用户环境中为所有支持 SIP 的设备动态地发现联系信息。通常情况下，也可以使用

其他服务发现协议。接下来将以 SLP 为例说明服务发现中会遇到的问题以及可能的解决方法。为完成以上所述，首先必须定义网络中各 SIP 终端上的“SIP”服务类型，使用标准的 SIP URL 来定义终端上的每一个服务，例如 sip：audio@devl. example. com。SLP 服务模版中必须说明用来指定设备特征（如厂商、支持的媒体类型和编码等）和位置参数（如房间号码）的服务属性。

在小型网络中，服务信息一般存储在 SLP 服务代理（Service Agent，SA）中，SA 既可以和 SIP UA 位于同一主机（如 PC）中，当 SIP UA 为 IP 电话这样的硬终端时，也可以处于不同主机。对于大型网络，SLP 可以选择使用目录代理（Directory Agents，DA），它使用集中式数据存储，为网络提供可扩展性和鲁棒性。在这种方式下，SLP SA 通过发送服务注册（SrvReg）消息向 DA 发送服务信息。注册之后，DC 上的 SLP 用户代理会更新所有在 DC 服务范围内设备的可用性。

为了能发现由多设备系统提供的服务，设备管理器必须首先通过向 DA 发送 SrvRqst 来发现其管辖范围内的所有可用设备。然后 DM 创建本地“用户名”来表示新设备（例如 sip：audio_video@ dc. example. com），并发送 SrvReg 消息注册此设备，注册消息里应包含系统的位置以及媒体属性。

当用户携带手机进入某个房间时，其设备上的 SLP UA 会首先发现 DA，这有两种方式：使用目录代理服务发送组播服务请求（SrvRqst）；通过动态主机配置协议（Dynamic Host Configuration Protocol，DHCP）来获取 SLP 选择项。一旦 MN 知道 DA 的位置信息之后，位于移动设备中的 SLP UA 会向 DA 发送“SIP”服务请求，并等待 DA 的回复，请求消息包含了所有 SIP 服务的 URL 和属性列表。为了过滤掉非本地服务的响应，MN 的位置属性信息应该与 SrvRqst 一同发送。

3.4　小结

服务平台体系架构在未来的移动通信系统中将越来越重要。不仅网络运营商为了增强自身竞争优势，会注重服务和应用的提高，而且服务平台也会向第三方供应商开放，以丰富服务种类。高级的商业模式，如前面提到的 i-mode，为参与方提供了双赢的局面，包括使用服务的用户，他们有更大的服务选择范围，而且服务也越来越个性化。事实上，个性化正是服务平台发展中用来吸引用户的主要驱动力。从这些新服务平台的能力以及范例中，我们可以立即得出结论：泛在通信是一个新的领域，它可以提供各种不同的服务，而且大部分都不是由传统的运营商提供。为了提供无缝的服务体验，在未来的服务平台设计中应考虑异构的泛在环境。

本章我们介绍了服务体系架构的相关内容，并重点介绍移动服务平台。不同的平台架构表明了服务提供的概念和范围，也强调了服务平台的重要性。基于最新的讨论，我们勾画出未来移动服务平台应具备的能力。特别是在以用户为中心的服务提供中，一定要考虑个性化和上下文感知。泛在服务应用在泛在通信环境中，并扩展了基于蜂窝网络的传统移动服务，提供趋于泛在通信环境的服务。为更好地说明会话移动性在服务平台中的实现，我们给出了在IP服务环境中使用SIP服务体系架构的详细设计。

第 4 章　泛在性扩展：移动 P2P

Zoran Despotovic and Wolfgang Kellerer

P2P 计算正逐渐成为一种重要的计算范式，将对服务平台产生颠覆性的影响。通过使用 P2P 技术，能够建立一整套全新的应用，并能推动服务平台发展到一个新的领域，这在几年前是无法想象的。

近几年来，学术界和工业界主要参与者越来越关注对等（P2P）计算。P2P 技术被视为一种颠覆性技术，能帮助建立一整套全新的应用，并影响信息技术未来的发展。今天典型的 P2P 系统如 Gnutella、Edonkey 和 BitTorrent 都是 P2P 成功的范例：通过简单的软件实现互联网用户的文件交换。这些应用的普及引起许多研究机构的关注[1]，据报道 P2P 流量在整个互联网流量中占压倒性的比例。

尽管是由文件共享推进了 P2P 系统的发展，但 P2P 技术的希望和潜力远远超过文件共享这一简单应用。最近几年其他 P2P 应用迅速发展，如 P2P Web 缓存[2]，P2P 信息检索[3]和分布式存储系统[4]。P2P 计算范式也为通常在底层实现的通信协议提供了高效的应用层实现方式，如应用层组播和任意播[5-7]。Stoica 等[5]甚至使用以 P2P 为基础的互联网重定向架构提供移动性支持。最近，这些成功的 P2P 系统设计思想都基于分布式哈希表（Distributed Hash Table，DHT），并为移动 Ad Hoc 网络的路由协议提供了新的思路[8]。总而言之，在经历了几年由于文件共享系统的版权问题而导致的 P2P 前途不确定性争论之后，P2P 技术在解决这一问题上，将对信息和通信技术产生重要影响。

P2P 系统在很多领域都提供了相对于传统客户端/服务器系统的替代解决方案，尤其是在互联网规模的分布式环境中。在客户端/服务器环境中，资源集中于少量服务器上，服务器为大量的客户服务。因此，必须建立成熟的负载均衡和容错算法来提供连续可靠的接入。另外，网络带宽必须随着用户数的增长而不断增加。相反，在 P2P 系统中每个节点同时充当服务器和客户端，为整个系统的成功运行贡献部分资源。换句话讲，每个节点通过提供自身计算资源的接入支付参加交换社区带来的收益。因此，P2P 方法解决了很多客户端/服务器系统的问题，包括可扩展性、容错性和管理代价等重要问题。

在本章我们将首先给出 P2P 系统的定义，其说明是基于 Aberer[9] 和 Despotovic[10] 的。出于为以后的讨论建立一个良好基础的考虑，这个定义相对技术化，

但我们认为它也足够简单和容易理解。接下来我们将讨论两类重要的P2P系统：非结构化P2P系统和结构化P2P系统（分布式哈希表）。我们在结构化P2P系统上讨论的较多，因为它们的性能优于非结构化P2P系统。在本章第二部分中，我们将详细介绍P2P技术应用于移动环境的情况。我们首先介绍一些潜在的移动P2P应用，然后总结将P2P技术应用于移动环境中面临的主要挑战，最后介绍已有的一些解决方法和开放的方案思路。

4.1 P2P系统的定义和分类

简而言之，P2P系统是一个应用层资源查找系统，其中参与节点之间相互建立逻辑连接以便实现对所需资源的有效定位。在详细介绍之前，我们先解释P2P研究的动机，明确我们为什么需要这样一个应用层查找系统。答案非常简单：在协议栈底层没有合适的支持资源定位的协议（这里“资源”指的是应用层概念），例如，文件共享系统中的文件就是一个“资源”。另一个例子是与一组计算机连接的传感器接收到的数据，在互联网中分布的大量计算机中的数据都是资源。很显然，IP可以在不同计算机之间路由，但IP无法知道什么数据存储在哪个计算机中。另一方面，事实证明基于产生和传播海量数据的大规模分布式应用有巨大的用户群。因此这类应用需要应用层资源查找系统作为主要的功能模块。

如上所述，P2P系统的核心问题是在没有集中控制条件下的资源定位。假设有分布于节点集P上的资源集合（我们用$p \in P$表示节点p的物理地址），R表示资源集合。假设资源使用应用层标识空间的键值K表示。假设距离函数d: $K \times K \rightarrow R$表示任意两个特定键值之间的距离。如前所述，资源的种类很多，可以是共享的文件、多样的数据，甚至是计算机的一个计算周期。键值可以是标识资源的正整数或字符串。

这样，问题可以表述如下：任意参与系统的节点应该可以尽可能快地定位到一个物理地址为$p \in P$、存有资源标识为$k \in K$的资源$r \in R$的节点。为了解决这一问题，节点根据一定原则，选择在少量其他节点建立逻辑的、应用层的连接并可向这些节点转发资源请求。也就是说，节点之间建立一个应用层的叠加网络（Overlay Network）。在叠加网络中进行资源寻址时应使用与应用相关的键值，而不能使用与应用无关的物理节点地址。该问题变成如何有效地实现这样一个叠加网络，即如何实现叠加网络维护和路由（比如在文件共享应用中共享新文件和搜索文件）的基本操作并保持物理资源消耗（如网络数据传输速率）较低。

我们概括P2P系统的主要组成部分如下，并简要总结上述推论：

一个P2P系统是一个元组（P、R、K、G、RA、MA），其中：

P：节点集合；

R：分布在节点 P 上的资源集合；

K：标识资源的键值集合；

G：表示叠加网络的有向图（P，V）；

RA 和 MA：图 G 上操作的算法，资源查找（RA）和网络维护（MA）。

假设已经建立一个叠加网络 G，路由算法 RA 可以看作一个转发的映射：$P \times K \rightarrow 2^P$，forward（$p$，$k$）是网络 G 中通过外连边 p 联系的节点的子集，其中 p 表示被请求用于定位资源 k 的节点。我们要强调图 G 中节点的出度不能过大，由于算法 MA 用于在节点加入、退出和故障时保证外连连接的一致性，过大的节点出度将导致维护代价过高。

根据上述描述，可以将 P2P 系统分成两类：

1）非结构化 P2P 系统：非结构化 P2P 系统的最大特征是资源在节点上的分布与叠加网络 G 的结构无关；节点间可以任意建立连接，与节点负责的资源无关。因此，资源查找以耗尽性方式（exhaustive fashion）进行，比如使用广播或洪泛方式进行资源搜索。非结构化 P2P 系统的例子有 Gnutella[11] 和 Lv[12] 等。

2）结构化 P2P 系统（也称为分布式哈希表，简称 DHT）：与非结构化 P2P 系统不同，结构化 P2P 系统中节点负责的资源集合与节点连接的其他节点集合之间存在相关性。通常，每个节点负责一组“邻近”的资源，与之建立连接的节点中，大部分是“附近”节点，小部分是“远端”节点。（注意所有“邻近”、“附近”、“远端”等词汇都与上文提到的距离函数密切相关）。在结构化 P2P 系统中，资源查找可以通过贪婪算法（Greedy fashion）进行，即向距离目标尽可能近的节点转发请求。结构化 P2P 系统包括 Chord[13]、CAN[14]、Pastry[15]、Viceroy[16]、Symphony[17]（基于 Kleinberg[18] 的小世界图理论）、Kademlia[19] 和 PGrid[20] 等。

图 4-1 总结了非结构化 P2P 系统和结构化 P2P 系统的不同定义。注意某些研究者认为混合式系统（参见 sGarcés-Erice 等[21] 和 Mizrak 等[22]）是第三类 P2P 系统。本书并未将混合式系统分为单独的一类，主要因为它们并未在系统设计中引入新的维度，而仅仅是上述两种系统的结合。我们将在 4.3.3 节中讲述混合式系统在异构系统环境中的优势。

此外，我们也没有在图 4-1 中显式包含集中式系统（如 Napster）。这种系统依赖一个集中的索引进行资源查询，而资源交换则采用完全的 P2P 方式。

接下来的两节将详细介绍上述两种 P2P 系统。考虑到 DHT 系统与非结构化系统相比的明显优势，我们提供了更多关于 DHT 系统的信息（这些优势将在下面的章节中有更清晰的描述）。为了详细比较 DHT 系统与非结构化系统以及比较不同的 DHT 系统，我们将着重比较以下几个方面：

1）叠加网络的结构如何，查找如何进行？

2）路由复杂度如何，即：查找的效率如何？

3）网络维护开销如何，节点加入和退出网络的效率如何？

读者可以查看一些最近的综述性文章[23-25]来了解关于这些问题更详细的分析以及一些其他重要问题的分析，如负载均衡、安全性以及基于邻近度的路由等问题。

客户机/服务器	P2P						
	用于资源共享的分散化、自组织的叠加网络						
中心服务器	非结构化P2P		结构化P2P				
	叠加网络结构不确定		协议确定叠加网络结构				
	扁平	等级式		环状	超立方体	树	…
	无中心实体 洪泛式搜索	等级方式的多种P2P概念的结合，如，结构化和非结构化；节点动态选举为负责叶节点单位更高级实体(超级节点)		节点和内容索引映射到一维DHT结构	多维结构，如：d维环空间	虚拟树结构	
如：万维网	如：Freenet，Gnutella 0.4	如：JXTA，Gnutella 0.6	如：图4.3	如：Chord	如：CAN	如：P-Grid，kademlia	

图4-1　Peer to Peer 概念

4.1.1　非结构化P2P网络

与结构化P2P系统相比，非结构化P2P系统最大的特点是节点不限制自身管理某一特定资源集合，从而资源在节点上的分布和叠加网络 G 的结构是独立的。因此任何节点都可以存储任何资源。也就是说节点独立地选择它们存储的资源，随机地连接到其他的一部分节点，而不考虑其邻居节点的资源构成。因此，为了获取其他节点管理的资源，节点使用图 G 来执行请求—转发策略，此策略必须确保节点最终可以到达P2P网络内的任意节点。

Gnutella是一个典型的非结构化P2P网络。在Gnutella中每个节点连接少数邻居节点，典型数目为4个。资源查找通过在叠加网络的大部分节点中进行洪泛查询请求完成，该请求起始于请求节点，递归式转发至所有已知邻居、邻居的邻居，依次类推。为了控制消息数量，洪泛设定一个上限，即所谓存活时间

（Time-To-Live，TTL），TTL 的典型值为 7。此外，为了避免消息出现回路，请求消息包含节点标识。由于 TTL 典型值的设定目的是遍历整个叠加网络，广播产生的消息数量与网络规模成线性比例（规模表示边的数目而不是节点数目），这也意味着在处理一个搜索请求时，几乎所有的节点对都需要交换消息。显然，这样的通信开销将非常大，这是非结构化 P2P 网络面临的最严重问题。但从另一方面讲，我们知道搜索延迟（这可能是用户体验中最重要的指标）依赖于叠加网络的直径，对于以 Gnutella 为代表的随机图和小世界图，其网络直径与节点数目的对数成正比，这意味着 Gnutella 能保证较小的搜索延迟。

在 Gnutella 中，网络构造和维护也使用相似的机制。加入节点使用洪泛方式发送发现消息（Discovery）。根据其他节点反馈，待加入节点选择其邻居节点。如果网络中某一节点失效，它将被网络中的已知节点列表中的节点动态地替换掉。研究显示这一机制将大致形成一个入连接幂律分布的网络，因为节点倾向于选择稳定的节点进行连接。类似的行为出现在许多自组织网络中，包括 Web。

上文已经讨论过，洪泛的代价相当大：一个单一搜索请求将引起一大群节点的负载，这可以解释为什么会有那么多的工作在讨论通过改善查询请求转发方式，来减少产生的消息数量，从而降低查询代价。一种思想是基于随机行走策略（Random walk）：采用深度优先搜索策略，请求不向所有邻居转发，而随机选择一个节点转发。另一种思想应用过滤理论来准确估计遍历整个网络所需连接数。尽管这种方法优于 Gnutella 性能，但其搜索效率仍然非常低。本质上说，类似 Gnutella 的系统缺乏节点间协调，也就是缺乏哪个节点管理哪个资源的信息。正是这个问题导致非结构化 P2P 网络的本质特征是通信开销大。

然而，非结构化网络也有一些非常好的特性。最明显的是更新一个资源和加入一个新的资源的操作代价非常低，甚至没有代价。比如，当新的资源加入系统，资源拥有者无需通知其他任何节点该信息。维护网络结构时操作代价也非常低。当节点离开系统时也无需通知其他节点。这正是节点和资源存储相互独立带来的优点，操作可以自主地完成。

另一个非结构化网络的优点是其处理复杂查询的能力，这一点我们在深入研究 DHT 网络的特点后可以有更深的体会。比如，如果资源有一组属性，本身就支持多属性查询，然而这在 DHT 中很难实现。

4.1.2　结构化 P2P 网络——DHT

与非结构化系统不同，结构化 P2P 系统中存在节点间的协调。本节将描述如何建立这个协调以及协调带来的好处和代价。

在 DHT 中每个节点负责管理一个特定资源子集。一般说来，某一给定时间内所有在线节点集合均按照如下方式决定资源的分布：每个节点首先关联键值空

间 K 中的一个键值（K 包含资源标识）。我们可以认为在这一步骤中节点随机选择键值。通常键值空间的大小远远大于节点数目，因此不会有两个不同的节点选择同一键值。同时，每个键值与键值空间的一部分关联，因此与该键值关联的节点负责管理该键值空间的所有资源。典型情况下使用恰当的距离度量方法，所有节点键值更接近（相对于其他在线节点）的键值形成同一键值分区。这就是为什么键值空间 K 需要配备距离函数 d 的原因。上述过程可以通过两个函数来表征：函数 key：$P \rightarrow K$ 将节点与键值关联；给定 key（P），函数 partition：$K \rightarrow 2^K$ 将节点与 K 的分区关联。

为了有效转发资源查询请求，节点根据节点与键值分区的关联关系组成一个路由网络 G。该网络可以被定义成一个另外的映射：neighbours：$K \rightarrow 2^P$，将图 G 中节点和它们的邻居进行关联。距离函数 d 在创建网络中起着重要作用。典型情况下，节点维护与其距离更短的节点的连接和邻居键值；同时也维护一小部分与其距离较长的节点连接，其中与一个长距离节点建立连接的概率随着到该节点键值的距离增加而降低。在多数结构化 P2P 网络中节点一般与长距离节点建立连接的概率随距离成指数递减。在 P2P 中，所有与节点连接的其他出度节点构成该节点的路由表，节点使用路由表直接转发资源请求到最近的节点，尽最大可能地减少到所查询键值的距离。

接下来我们以一个具体的算法来介绍 DHT 的工作机制。Chord 是最流行也是最简单的 DHT 算法，如图 4-2 所示，键值是按顺时针方向排列在环上的整数（与笛卡儿空间上的字符或点相对）。环上最顶端的点对应键值空间大小，所有键值都与该数字取模后加入到环上适当的位置。距离 d 定义为两个节点间的顺时针距离。节点负责的键值分区起始于自身，终止于逆时针下一个可用节点（但不包含该节点）的弧形区域。如图 4-2，按照这种定义方式，节点 P14 负责键值 14、13、…、9。图中还显示节点 P08 的路由表，节点 P08 维持与其键值的距离为 2 的 N 次方的节点的连接。因为其后继节点会负责键值 $8+2^1=9$[㊀]，因此，P08 会维护与其直接后继的连接。同理，节点 P08 也会和负责与其键值距离为 2（2^1）和 4（2^2）的节点建立连接。最远距离连接的节点键值位于相距该节点半个键值空间处。因此 P08 与键值 40（$8+2^5$）所属分区的节点 P42 之间有一个连接。如果节点以上述方式相联，路由将变得非常简单：请求转发给不超过目标键值的最远连接；可能超过该键值的节点选择其直接后继节点作为目标节点。比如，节点 08 要路由到节点 P54，它首先利用最远可能连接到达节点 P42，P42 接着转发请求到 P51（再远将超过目标键值）。最后 P51 将请求转发给其后继节点 P56。

㊀ 此处应为 $8+2^0=9$。

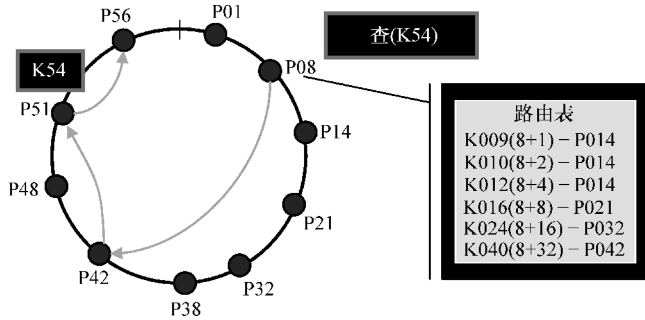


图 4-2　Chord DHT（经作者和 ACM 出版社同意根据文献［13］重新绘制）

Chord 是使用确定性邻居函数的范例。由于使用确定性邻居函数，如果给定参与节点，叠加网络的结构预先可知，与此相对的是随机系统。

显然，Chord 的路由长度与网络大小（即网络中总节点数）的对数成正比。因为每一跳使到目标节点的剩余距离减半。同时，路由表的大小也与网络大小的对数成正比。毋庸置疑这在大多数情况下可以接受。对于拥有几百万节点的大规模文件共享网络，这意味着任意文件可以在几步内找到，而每个节点只需要维护一个较小的路由表。这两个特征对多数 DHT 系统都是成立的，但是需要指出的是，最近有一些工作实现了常数度拓扑和常数规模路由表，同时仍保持对数级别路由。两个知名的例子是使用蝶形拓扑的 Viceroy[16] 和使用德布鲁意图的 Koorde[26]。

一般说来，任何 DHT 方案都可以通过描述其键值空间选择、距离函数和连接策略来表示清楚。经过几年的深入研究，现在已经出现了很多种结构化叠加网络的设计。表 4-1 总结了一些代表性系统的主要特征。注意一些系统设计在路由复杂度和路由表大小上进行了折中。比如，Koorde 路由表大小为 O（log$N$），其路由长度为 O（log$N$/loglog$N$）。

表 4-1　不同 DHT 的主要特征

P2P 系统	比较指标				
	几何拓扑①	距　　离	参　　数	数据查找跳数	路由状态
Chord	环状	顺时针整数差	N = #peers	O（log N）	O（N log ）
CAN	超立方体	d 维空间中的笛卡儿距离	N = #peers	O（$d \cdot N^{1/d}$）	O（d）
Pastry	混合（树-环）	包含源、目的的最小子树深度或者顺时针整数差	d = #dims	$\log_{2^b} N$	$b \cdot \log_{2^b} N + b$
Kademlia	树	异或值	N = #peers	O（log N）	O（log N）
P-Grid	树	包含源、目的的最小子树深度	b = id base	O（log N）	O（log N）

（续）

P2P 系统	比较指标				
	几何拓扑[①]	距　离	参　数	数据查找跳数	路由状态
Viceroy	蝶形	顺时针整数差	N = #peers	O（log N）	O（1）
Smphony	环上的 kleinberg 小世界	顺时针整数差	N = #peers k = outdegree	O（($\log^2 N$）/d）	O（d）
Koorde	德布鲁意图	顺时针整数差	N = #peers	O（log N）	O（1）

① 如 Gummadi 指出，几何拓扑（Geometry）的概念不够准确。本书中我们按照读者的习惯使用这一术语。

4.2　DHT 的一些问题

在本节我们将详细阐述一些 DHT 中看似简单但实际很复杂的问题，说明了构造一个高效 DHT 路由所要付出的代价。首先，作为引入节点之间关联的必然结果，在节点加入、退出时必须进行网络维护来保持路由表的一致性。保持小路由表的原因实际上并非是存储代价，而是大路由表带来的维护开销。第二，与非结构化 P2P 网络不同，资源本身的变化，即资源的插入、更新和删除并非只与维护该资源的局部节点相关，而且要影响其他节点。第三，资源请求被仅限于简单的基于键值的查询，即使简单的普通查询，如基于键值范围的请求也需要对算法进行重大改动。

4.2.1　维护开销

如果我们基于上面讨论的内容判断，DHT 看起来是大规模 P2P 系统的一个完美解决方案，因为它可以实现路由的高效性，同时每个节点只需要保持很少的节点状态信息。然而，高效的路由并非没有代价。为了建立和维护一个结构化的 P2P 网络，节点不得不特别处理节点的加入、离开和故障的问题。

节点加入的过程常与路由操作重叠，因此算法复杂度也是对数级的。在加入系统时，节点通常通过它们预先知道的节点路由到它们自己的键值。通过这种方法节点可以知道它们在 DHT 中的位置以及可以用来构造它们自己的路由表的其他节点路由表信息。作为加入的最后一步，节点接受其负责资源集合的索引，这些索引通过与加入节点键值最近的节点集合获得。节点离开的操作与此类似，主要目的也是维护网络的一致性。比如，当一个节点离开时，它应该通知那些指向它的节点其离开信息，以便这些节点更新路由表。因此，加入和离开网络都会产

生开销，通常开销不太大但也不能忽视。

另一个产生开销的原因是 DHT 节点可能出现故障。这与节点离开不同，因为它们在离开前不事先通知，因此难以保证网络一致性。问题是网络不一致可能导致 DHT 性能的严重降低，在最坏的情况下网络可能出现分块，可能存在节点群不连通。这就是为什么节点必须执行额外的算法以保证网络一致性和网络分块时的重建的原因。这通常涉及节点探测器路由表中的所有表项以及用一个活跃节点代替一个失效节点。如果节点故障率很高，上述步骤必须频繁执行，这可能放大失效率以致产生不想出现的结果。因此，必须仔细设计故障检测和恢复算法。读者可以参考 Rhea[27] 的文献得到本问题更深入的分析。

4.2.2　复杂查询

DHT 本质上只支持精确查找：为了搜索某资源，必须知道与资源关联的键值。然而在大部分情况下请求者并不知道确切的键值，因此单单基于键值的搜索是不够的。这严重限制了 DHT 的实际应用范围。典型情况下，每个资源标识为一组属性-值对，用户希望收集所有属性集合内特定值的所有对象，并能够执行更复杂的查询。比如，在文件共享 P2P 网络中许多搜索请求包括文件名的一部分、文件类型（如音频和视频文件）以及演奏者的名字（对于音频文件）等。DHT 不支持这类查询[28]。

所谓范围查询，就是通过键值或其他属性值返回特定数值区间内的所有对象。其中资源的键值是从各自的属性值中衍生计算出来的。比如，键值可以通过对资源代表的二进制数（一部分）进行 SHA-1 哈希获得。在这种情况下，语义上相近的对象，即那些在一个或多个属性值上只有微小差别的对象在哈希值上完全不同，因此将以非协调方式散布于网络上。之所以使用这样的哈希函数，是因为设计者希望实现负载均衡：节点应尽可能均匀地分担负载，也就是它们应当负责大致相等数目的键值。另一方面，使用保序哈希函数，可以保持一个属性的多个值的顺序。但这将破坏负载均衡，因为属性值的偏态分布将导致键值的偏态分布。

因此负载均衡和支持范围查询两个问题无法兼顾，解决其中一个问题便会导致另一个问题的恶化。P-Grid[20] 对范围查询的支持性较好，它使用保序哈希函数和其他系统不同的负载均衡策略。在 P-Grid 中实现范围查询的算法可以在 Datta[29] 的文章中找到。Triantafillou 和 Pitoura[30] 使用类似的思想实现了范围查询，同时使用复制策略（在实际系统中用于高容错性）实现负载均衡。资源通常有多个属性，负载均衡通过不同属性使用不同哈希函数获得。

但是多数系统的做法与此相反，它们使用上述方式实现负载均衡，同时提供额外的对范围查询的支持。Ramabhadran[31] 等通过在已经存在的 DHT 上添加对

一个选定属性原始值（未哈希）进行“前缀哈希树（Prefix Hash Tree，PHT）”的方法实现范围查询，试图以此重建由于哈希打乱的顺序。与P-Grid天然支持范围查询相比，PHT效率较低，主要原因在于给定一个查询数值，由于PHT不维护相关信息，需要搜索与该值对应的多个前缀。我们假设对于不同前缀进行二进制搜索，属性空间的大小为2^D，则PHT操作的次数为O（$\log D \cdot \log N$）。但需要指出的是PHT无需指定底层的DHT（可以在任何DHT上实现），而且实现了范围查询，这是我们认为其最主要的优势。

对于更复杂类型的查询，比如多属性查询，通过事先设定一组属性的值，返回满足多个条件的所有对象，使用什么样的方案（我们仍假设一个简单的数据模型：每个资源与多个属性—数值对关联），最通用的方法是为每个属性建立索引，然后为请求中的每个属性运行查询过程。最终的请求处理，即过滤返回的结果，可以在请求节点本地完成。尽管分布式请求处理在数据库研究领域研究得相当深入，但很少有将研究结果应用到优化P2P请求处理策略中。而且，选择合适的属性作索引非常重要，从来不被请求或很少被请求的属性不应被索引。更细节的讨论，读者可以参考Klemm[32]。

将我们刚才讨论的想法推广到一般化，可以应用于系统中可用资源包含更复杂结构的情况下，尽管这时问题变得更加复杂。不失一般性，假设每个资源与一个XML描述相关联。这些描述可以相当复杂，它们的构架（Schemas）也不必相同。在这种情况下一个标准的方法是生成一个辅助的索引（元数据）描述返回结果的请求，即在用户执行的“概括”查询和返回的对象结果间建立影射。这个辅助的索引作为补充数据集存储于底层DHT中，其中以请求为键值。请求处理分为两个步骤：给定一个请求，首先请求辅助元数据，找到将出现在结果集中的对象键值；接着使用这些键值进行DHT请求，以找到这些对象的位置。这实际超过两步操作，我们可以从比较概括的请求入手，逐步细化到比较详细的请求，直到确定的请求。Garcés-Erice等[33]以及Skobeltsyn等[34]提出两个使用这种方法的例子。尽管这个思想提供了解决复杂查询问题的一个方法，但这并非完备的方法。最严重的问题是即使是简单形式的数据，也很难预测对这些数据的确切请求。从理论上说，可能的请求数目非常之多，即使可能，维护包括所有请求的索引列表会成为一个负担。这就是为什么预测请求分布变得非常重要，即什么请求可能会以多大频率被用户使用。一旦找到合适的方法，问题便为如何降低维护索引的代价以及如何在索引带来的好处和引发的代价间找到最好的折中。

4.3 移动P2P

现在讨论如何将P2P思想应用到移动环境中。我们将简要概括P2P技术给

移动用户、运营商和潜在的服务提供者带来的好处，同时也要指出很多实用的应用可以使用 P2P 方式实现。在本节第二部分我们首先讨论通过 P2P 方式实现此类应用所带来的挑战，然后总结可能的方法。

在我们深入讨论之前，首先介绍移动 P2P 网络的一个简单且有用的特征：它是用于交换数据的移动通信系统，至少有一跳通信经过无线连接。与传统的固定网络相比，移动 P2P 系统的一个重要特征就是系统异构性。

4.3.1　P2P 技术给移动用户、运营商和服务提供者带来的好处

随着数据通信系统可用性的提高（包括接入移动网络可以进行互联网），始于有线网络的 P2P 应用也可以在移动网络中使用。用户希望能够在移动时享受与有线环境同等质量的服务。

这只是移动网络可能出现一系列新应用的一个原因。另一个主要原因是 P2P 技术加速特定应用的发展并使之更容易使用。更精确的表述是，一旦提供一个 P2P 平台（包括路由、数据管理层等），在此之上建立新应用将非常容易；在许多情况下它只涉及细化数据管理功能以及选择合适的数据表示以满足应用和性能的要求。这可以解释为什么移动运营商对提供这样的 P2P 平台兴趣浓厚，他们将 P2P 平台作为鼓励第三方开展新服务的重要手段。目前第三方服务提供者在移动服务中各不相同，但都可以在移动服务中提供新的价值。因此，我们相信第三方甚至可能是移动用户自己。不管情况如何，我们都可以看到一个全新的市场，用户自己可以提供各种各样的服务，相互可以交易。基于用户本地邻近信息的服务就是一个有趣的例子。用户可以与邻近用户交换不同类型的信息，如免费停车、好的餐馆、特定区域的旅游景点、当地商店打折信息等。

此外，P2P 计算模式可以扩大运营商的服务范围到泛在环境（也就是具有不同类型的空中接口的松耦合系统），同时系统无需像现在的蜂窝系统一样使用集中式服务器。可以促生一系列使用和处理来自传感器网络等的数据的新应用。移动电话可以使用这种应用为用户提供一个泛在的接口或作为移动传感器，在下一节我们介绍这样的应用场景。

最后，P2P 系统不依赖于基础设施而是依赖于用户的设备提供服务。基于可用的用户设备而不是使用昂贵的基础设施提供移动通信降低了维护成本和服务提供成本。

4.3.2　移动 P2P 应用

原则上说，大多数传统的 P2P 应用都可以在移动 P2P 系统中实现，以提高移动网络用户容量和服务种类。此外，移动网络有能力承载新的应用，如基于位置的应用、建立自组织社区以及泛在环境下的各种应用。

一般说来，P2P 应用的分类包括数字信息共享、个人通信、社区和协作服

务、叠加网路由重定向（如组播）、网络管理、流媒体以及网格服务等。除了众所周知或正在兴起的P2P应用如文件共享或P2P VoIP，P2P的机制也可以通过在IP层上面引入一个新的转发层支持一般的互联网服务，如组播、任意播以及通过P2P叠加网络实现移动性[5,35]，传统的网络管理也可以从P2P技术中受益。P2P流媒体应用与P2P数据共享概念不同，媒体可在下载时直接播放，建立起一个不断增长的流媒体节点链。分布式服务提供是P2P思想的另一个应用，比如Web服务可以从客户机/服务器模式改为P2P网格模式。在接下来的章节中，我们详细讨论移动P2P相关的应用。

1. 信息共享

文件共享是P2P应用最突出的例子，可以在用户之间共享数字化信息。然而，这只是一个例子。这种应用类型可以扩展到共享任何用户编辑的信息，包括Web logging或博客这类非常流行的应用，在这些应用中用户可以在线发布他们的评论和信息。我们强调在移动环境中，这类P2P应用可以通过位置信息实现功能增强，从而更加有用和流行，位置感知意味着根据请求者的位置提供与之最相关的信息。传感器网络中的用户信息和环境上下文也可以从P2P查询机制中受益。

2. 社区应用和组通信

P2P技术支持无需额外基础设施条件下（如一个呈现服务器），能基于位置或兴趣自发建立虚拟社区。应用包括上面介绍的基于主题的（如世界杯）、基于位置的（一个体育馆或音乐会）或基于其他一些共享特征的（如堵车时候汽车之间）简单的信息共享。自发社区的共享服务不仅限于信息共享，还包括任意类型的服务，如卖票或P2P游戏。依据服务类型，可能需要额外与服务沟通、合同执行相关的支持。现在，已经能够实现这类应用最主要的功能。读者可以参考Castro[36]的P2P组通信解决方案、Aekaterinidis and Triantafiuou[37]的基于内容的发布—订阅系统。

3. 个人通信

不仅仅是最近Skype（见www. skype. com）的成功才使人们激起对基于P2P的个人通信的兴趣（见www. p2psip. org）。除了VoIP，即时消息系统也使用P2P技术搜索用户地址，从而取代集中式注册服务器。由于无需运营者维护基础设施，P2P技术节省了大量的成本。移动应用同样可以得益于位置感知和自发、无基础设施的社区建立。存在的挑战是如何提供基于服务器的蜂窝电话应用，如语音信箱。

4. 泛在环境

本书已多次提及，通过增加通信能力许多设备的功能得以增强，比如家庭网络、无线传感器网络以及远程信息处理。在不久的将来，一个传感器、人和各种不同对象共存、移动、相互通信的泛在通信环境即将出现。我们相信P2P技术将在这种环境中起到重要作用。想象如下的场景：商店的顾客需要商品的产地、价格和有货缺货信息，以及其他顾客的意见。商店可以通过装置的传感器获得大

部分的有用信息。为了让这种信息在全球范围共享（如全球性商场），商店可以运营一个 P2P 网络。移动用户是该网络的有机组成部分。他们在踏入商店的第一步就加入到该网络并使用该服务，在离开商店时退出该网络。从技术上讲，我们不希望移动用户主动存储可用数据或路由转发请求，因为他们的短暂在线时间可能引起问题。在接下来的章节我们讨论适应上述需求的 P2P 结构。

4.3.3　移动 P2P 面临的挑战

在移动 P2P 中有很多挑战要解决。下面给出我们发现的非常重要的性能需求列表（更多讨论参见 Kellerer[38]）。注意本列表中的最后一项要求：信任和激励模型，极其重要，它不仅能提高所选 P2P 方案的性能，而且可以提高用户接受应用程度，因为一般在应用层和 P2P 路由层使用相同的模型来优化用户行为。

需求列表包括：

1）尽可能降低 P2P 查找流量的开销，以适应移动设备低传输速率的环境。

2）解决由于节点频繁加入、退出引起的高扰动性问题。

3）考虑移动设备资源受限和节点异构性问题以及它们特殊的设备能力。

4）考虑底层拓扑、最小化物理网络产生的搜索流量。

5）提供信任和激励模型，提高用户遵守协议的意愿。

本章第一部分已经讨论了第一项需求。中心思想是相对于无结构化系统 DHT 提供了相当好的性能。这就是我们为什么在本章倾向于使用 DHT 的主要原因。然而，最终决定哪个 P2P 结构适用于给定的移动环境不能仅仅依赖于查询代价，而应该考虑上述列出的所有因素。在接下来的讨论中，我们对一些现有的方法进行仔细评估。在评估之前，我们首先考察在不同的移动网络中对这些需求的迫切程度，也就是说，底层移动网络的特性如何影响这些需求。我们将看到，在一些网络中需求不那么紧迫，但在另一些网络中需求则相当紧迫。我们这里根据限制条件从小到大考察了蜂窝网络、热点网络、Ad Hoc 网络以及传感器网络。关于无线传输系统更详细的技术讨论参见第 2 章。

蜂窝网络是一跳的无线网络，通过单条无线连接将节点与固定的互联网相连。移动性管理使得切换等移动性问题对应用透明。在蜂窝网络中，主要技术难题在于吞吐量和时延等数据传输速率的限制以及设备资源的限制。比如，通过 9.6kbit/s 的 GSM 以及低于 50kbit/s 的 GPRS 等第二代移动通信系统连接的节点只能被视为低性能节点。第三代移动通信系统如 UMTS 在欧洲可以提供384kbit/s 的下行链路。但是在已部署的大多数系统中上行链路只有 64kbit/s。手持终端通常只有有限的存储空间、处理能力和电池。尽管它们只能提供有限的处理能力，但大多数的数据格式，如音频和视频仍可处理。而且，更加强大的设备如笔记本和 PDA 等也可以使用手持中继站或数据卡接入。移动电话通常为了获得可用性

不断进行切换。而且，在蜂窝网络中的高昂传输代价使得扰动率仍然是个问题，它会导致用户关闭其数据应用程序，这在2G系统中尤为严重。由于2G和3G系统的同时使用，异构性也是一个问题。

无线数据网络热点在有限范围内提供了相对高的传输速率。基于IEEE 802.11b的无线局域网今天大多拥有10Mbit/s的速率。尽管终端能力和速率对于P2P系统已经足够，然而移动被限制在一个热点区域内。也就是说，移动性仍然是一个问题，因为用户移动将引起网络的高扰动。

无线局域网技术在移动Ad Hoc网络中也很普遍。在移动Ad Hoc（MANET）网络中，节点间的传输路径通过使用中继节点进行多跳无线链路实现。由于移动Ad Hoc网络中假设节点是移动的，临近节点移出射频范围导致的链接中断，从而引起高扰动的现象非常常见。同时，MANET路由带宽消耗较大。这个问题在没有考虑底层网络拓扑的P2P叠加网络中更加严重。因此，距离远近感知在基于Ad Hoc的P2P网络至关重要。在不同的MANET中，节点能力千差万别，因此资源受限问题也很关键。

Ad Hoc传感器网络可以被看成是拥有极端受限资源的MANET，只适合特定类型的P2P应用。

4.3.4 异构移动环境下的移动P2P叠加网络

移动环境通常包含异构的设备、从手机到PDA或笔记本，接入方式也千差万别，包括GPRS、无线局域网或UMTS。现存的DHT方案大部分集中于所谓的扁平化P2P设计，也就是说，所有节点平均分担负载。但这个负载对一些低端节点来说，如手机来说显然太高。因此，或者不用这些节点，或者整个系统性能很差。这一问题引起在DHT结构中引入节点异构性的探索。这是我们为什么研究这种DHT结构的重要原因。通常，不同节点的通信开销也不同。比如，移动电话的通信开销比ADSL接入的PC更高。从直觉上讲，如果具有更高通信开销的节点执行更少的路由（也就是引起更少的开销）而获得同样的收益，则系统操作整体上更高效。

至于非结构化P2P方案，一般不存在异构性问题。低端节点搜索失效和高端节点退出网络都不会影响其他节点。然而需要指出的是，在非结构化P2P中也有混合式方案（如Gnutella0.6[39]）间接处理异构性问题。这些方案的主要焦点在于降低混合式结构中搜索的复杂性，但可以很容易地解决节点异构性问题。

我们介绍两个专门解决异构性问题的DHT结构。第一个是混合Chord协议[40]（Hybrid Chord Protocol，HCP），如图4-3所示。

HCP区分两类节点，固定节点和临时节点。固定节点具有高可用性、高速连接、高处理能力以及高存储能力。假设它们在叠加网络中具有较长的生存时

间，因此可以存储所有的对象索引。临时节点仅短暂加入叠加网络。它们不存储资源对象索引，但参与路由。当一个临时节点负责一个请求时，它将请求转发给最近的后继节点（如节点 D 转发对对象 W 的请求给节点 E）。HCP 使用 Chord 环作为基本的请求路由结构，但它很容易适应于任意 DHT[41]。根据在线时长和性能选择固定节点，使用这一思想可大大减少由于维护 DHT 网络而带来的流量，因为频繁加入和退出的临时节点不会引起资源键值移动，而且临时节点（如手机）没有存储和解析键值带来的负载。

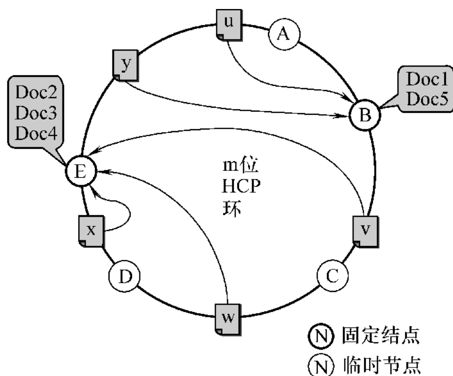


图 4-3　混合 Chord 协议

然而，这种方法没有给根据我们上面提示的代价模型优化网络操作留下太多的空间。实际上，即使是执行路由对一些节点来说压力也太大。在上面的例子中这对应于将临时节点移出 Chord 环。不同的混合式 DHT 设计这时就有了用武之地。图 4-4给出了一个非常简单的 DHT 结构，其中低端节点不参与 DHT 路由但仍参与该系统。这些节点（称为叶节点）与其代理节点（超级节点）相关联，代理节点加入 DHT 中。Zöls[42]提供了另一个支持使用该类 DHT 结构的证明。这种系统的总操作代价小于所有节点参与 DHT（假设低端节点能够承受这种负载）的扁平化 DHT 系统。我们发现理解这一问题的关键在于：混合式结构实际上是对集中式系统和完全分散化系统（扁平化 DHT）进行了折中，前者虽然能最小化操作代价，但由于没有一个节点能够承担整个系统的全部负载而无法实现；后者引起最大的操作代价，但在实际中可以实现。

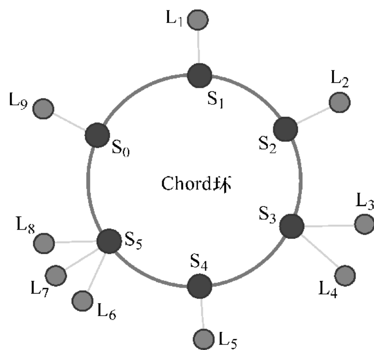


图 4-4　等级化 DHT

简单的混合式设计不是唯一的选择。Garcés-Erice 等[21]提到了一些其他可能的混合式设计，如将操作留给一个新的 DHT 而不是完全依赖于上层的 DHT 代理。这种方法可以在一定程度上降低上层节点的负载。我们认为，将 Zöls 的代价模型引入其他的混合式设计并比较不同设计将会是非常有趣的工作。

4.3.5　P2P 叠加网络和物理网络感知性质

到目前为止，从我们的讨论中似乎可以看出在叠加网络中的跳数是衡量 P2P 对象定位算法好坏的惟一标准。这一标准确实非常重要，但也对读者的理解产生了一些误导。最终根本要素是在物理网络引发的消息数量，而非叠加网络中的跳数。因此，即使通常在一般情况下叠加网络中较少的流量会对应于物理网络中的较少流量，但二者也不能割裂。显然，降低物理网络中的流量要求叠加网络能够感知物理网络及其拓扑以及其他性质。叠加网络中逻辑上靠近的节点应尽可能物理距离上也较近，否则整个搜索延迟会非常大并严重降低系统性能。

为了说明这个问题的严重性，我们假设在 MANET 上使用一个非结构化的 P2P 网络。在路由一个叠加网络消息时，节点需要发现目标节点的物理地址（依靠 MANET 的路由协议，也可以通过在物理网络上洪泛）。在这种情况下，与 P2P 网络相关联的整个路由代价即使可以接受，也会异常高。然而，如果基于对底层 MANET 的知识构建叠加网络，这个代价将大大降低，这样可以避免产生由于缺乏对底层网络拓扑了解引起的 P2P 叠加网络中的长路由或 Z 字形路由（Zigzag routes）。Gruber 等[43,44]提供了一个使用跨层设计原则的简单完善的解决方法，这一方法称作移动 P2P（MPP），如图 4-5 所示。MPP 使用 MPCP 实现层间通信，在网络层中插入请求。特别对于 Ad Hoc 网络中的动态源路由（Dynamic Source Routing, DSR）[45]协议，可以通过插入应用层 P2P 请求实现性能增强成为增强型 DSR（EDSR）。这样，请求被路由到网络层的邻居节点而非应用层的邻居节点，后者对网络距离无法感知。在每个节点上，请求通过 EDSR 被转发到 P2P 应用层并进行关键字匹配检查。如果没有成功，应用层再次插入请求进行继续转发。这比只在叠加网络上进行搜索更有效率。

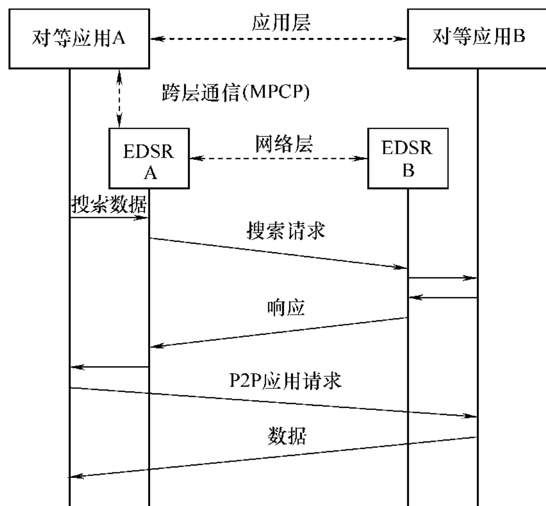


图 4-5　移动 P2P 协议

我们刚才讨论的问题既不是非结构化 P2P 网络的特有也非 MANET 网络的特

有。它在固定网络和 DHT 网络中也存在，研究界也展开了相关研究。现在大体上有两类解决方案，物理邻近节点选择和物理邻近路由选择，其目标是降低物理网络流量和搜索延迟。这两种方案的区别非常小。在物理邻近节点选择中路由表基于邻近关系构造而成；而在物理邻近路由选择中，依靠已经存在于路由表中的邻居邻近关系选择下一跳。Gummadi 等[23]研究证明物理邻近节点选择性能优于物理邻近路由选择。因此，在构建路由表时仔细选择节点变得非常关键。与此紧密联系的是路由表构造的灵活性——更灵活能选择更好的邻居。Gummadi 等[23]进一步研究发现，环形和树型拓扑（如 Chord 和 Pastry）的灵活性比较好，而超立方体拓扑（如 CAN）则没有足够的灵活性。有趣的是，拥有常数规模路由表和对数级路由的拓扑（如 Viceroy）则完全没有任何灵活性，他们完全不承认邻近物理距离的节点选择。

4.3.6　P2P 信任和信誉度管理

在 P2P 应用中，节点之间相互提供服务，用户通常希望能确保提供服务的节点可信。信誉度管理作为一种推进 P2P 网络可信行为的方法在学术界已经被进行了研究。P2P 信誉系统的关键在于搜集和散播可能影响未来交易的历史行为信息[46]。这里，一个隐含的假设是节点间的交互是天然可重复的。

P2P 信誉系统不仅在 P2P 研究界，而且在人工智能和经济学界也引发了研究兴趣。大多数关于 P2P 信誉系统的研究（如 Kamvar 等[47]以及 Despotovic 等[48]的研究）都假定节点按照某种静态概率分布，从而决定节点以特定的方式工作。这些方案的主要关注点在于依照节点的历史表现发现节点的这一分布，并提前通知与之交易的节点。因此，P2P 信誉系统的目标就变为交易中通告什么可以变坏、多大程度得变坏。目前还不清楚这些通告可以多精确地影响节点的未来行为。

另一方面，信誉度的概念已经在博弈论中成为一个研究分支（参见 Kreps 和 Wilson[49]的研究）。参与方的交互被看作是一个重复的博弈，意味着节点完全是理性的效用最大化者，最终希望最大化其长期收益（在整个重复博弈过程中）。节点在任一阶段采用哪种策略决定于它们和它们的交易伙伴之前的交互行为信息。仔细选择交互节点获得的这种可用信息的形式（反馈累积）可能直接影响节点将来的表现。理想情况下，选择可信的行为作为最优策略是有可能的（在博弈中体现为特定的行为）。

所有这些原则上都可以用到 P2P 路由本身，而不仅限于假设使用一个正确运行的 P2P 路由层来提供任何所需的信誉度数据的 P2P 应用。但问题的复杂性在于信誉度系统和核心 P2P 路由层之间存在的依存关系：前者需要后者保证其正确操作，反之亦然。我们不清楚是否有工作评估这样的系统，即：信誉系统运

作如何、会对路由产生何种影响。读者可以参考 Blanc[50]的文章，其中对如何构造一个路由博弈以及对信誉系统的影响等有更深入的讨论。

在我们看来，概率模型和理性模型都有些极端。它们与人们的实际行为不完全一致，因为人们既不是完全理性也不是完全概率性行为。我们相信，能够刻画学习历史经验但在整个生命周期中不完全优化的其他形式的行为值得探讨。包含不同互惠形式的简单模型就是一个很好的例子。

4.4 小结

P2P 系统现在已经发展成为一种重要的计算模式，对服务平台存在颠覆性的影响。它带动了一批新的应用，将服务平台推进到几年前无法想象的阶段。最引人注目的是，融合传感器、蜂窝和固定网络的泛在环境可以得益于 P2P 计算模式。

在本章的第一部分，我们描述了 P2P 的概念，分析了 P2P 系统的主要类型：非结构化 P2P 网络和结构化 P2P（DHT）网络。我们发现 DHT 在大部分情况下更合适，包括有移动用户参与的情况。这主要是因为其优良的可扩展性，如在大多数操作中的对数级复杂度。然而，我们也指出了 DHT 概念中的一些问题以及解决这些问题的代价。

在第二部分，我们讨论了如何将 P2P 技术应用于移动环境，列出了其中最重要的问题。我们发现大多数为固定网络设计的可用的 DHT 方案无法直接应用于移动环境。我们需要重新考虑节点异构型、节点有限的资源以及底层网络的限制等因素影响的 DHT 结构。最后，我们提示了一些目前这方面的方案。尽管这些方案证明学术界在提供移动 P2P 平台方面有一些进展，我们相信为了在移动环境中完全实现 P2P 技术的潜力，仍有大量工作要做。

第5章　移动中间件

Chie Noda

在分布式环境中，精心设计的中间件对服务和应用都提供了有益的支持。它隐藏了分布式系统的异构性，如网络协议、操作系统以及硬件等的不同。特别是在移动通信中，最近涌现的泛在网络和业务提出了对于运行于异构设备上的新型中间件的需求。“移动中间件”是专门用于移动网络环境中的中间件，以应对移动、泛在和资源受限设备的挑战。

传统的蜂窝移动网络（如GSM、PDC以及UMTS）使用网关服务器提供集中式服务，如i-mode服务以及多媒体短信。最近出现的固定性宽带无线技术（如WLAN、WiFi、WiMAX）以及固定移动融合技术构造了多运营商、多射频接入、多网络漫游的异构性环境。接入技术的分散化可能建立起一个集中式软件系统无法解决的开放、动态配置的基础架构。在这样一个动态、开放和复杂的服务环境中，集中于用户需求和行为的以用户为中心的计算模式正演进成为下一代中间件的重要特征。

随着移动设备的发展，人们正试验日益强大的移动和分布式计算环境。这些现象都加速了移动通信中的泛在网络服务的出现。移动设备有望支持短距离通信技术（如NFC和ZigBee），成为传统移动网络和即将到来的传感器网络的中间点。将移动设备和大范围的网络技术结合，使这些每天增加的、围绕在用户周围的泛在设备能够在这种环境中交互，查找用户环境信息（如位置和温度）和用户状态（如在工作、在度假、在开车）并进一步真正帮助用户[1]。

中间件定义为异构环境下开发、部署和管理分布式应用提供接口和服务的分布式平台。中间件技术就是一类帮助管理分布式系统固有的复杂性和异构性[2]的技术。它通过隐藏异构性（如异构的操作系统、硬件、网络协议和数据库）降低分布式系统开发者的负担。图5-1是一个中间件的层次图，它是位于操作系统和分布式应用之间的中间层软件。中间件通过数据通道与分布式应用相连并彼此之间交互数据。

移动中间件是适合移动网络环境的支持移动性和运行于资源受限移动设备上的特殊中间件。正如Gaddah和kumz等[3]讨论的，传统运行于宽带有线网络和资源不受限设备上的中间件一般并不考虑移动中间件的特殊要求，这些要求包括：

1）稀少资源：移动中间件分布于资源受限设备，具有较低性能，有限的内

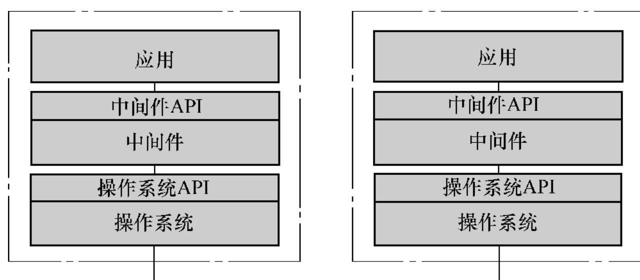


图5-1　中间件层

存，较窄的射频带宽以及低功率消耗需求。即使硬件演进遵守摩尔定律，与其他网络节点和应用服务器相比，在资源上和能力上仍存在较多限制。因此，移动中间件必须优化这些稀少资源。

2）异步通信：移动网络连接通常具有有限、变化的传输速率并频繁地掉线，如在移动设备移出网络覆盖范围或关机时情况。人们希望能有更高的通信速率，下行带宽通常假定远高于上行带宽。如果考虑这些特征，移动中间件必须支持异步通信，解耦客户端和服务器并提供可靠性。

3）上下文感知和适应性：服务质量（QoS）在异构无线接入环境下将动态变化。不仅仅QoS，而且包括通用应用上下文，如位置、用户上下文等都将动态变化。移动中间件需要支持上下文感知，即：感知上下文的变化，动态变化及其行为或将上下文信息转发到系统的其他部分。比如，当给上层提供上下文信息来帮助用户（详见5.3节）或给底层提供上下文信息来最优化网络参数（详见5.3节），系统可以支持以用户为中心的服务提供并增强这些系统性能。

接下来我们分4个方面讨论如何建立移动中间件：

1）移动中间件技术作为传统平台的扩展，支持移动性和资源受限。

2）中间件结构中需要的功能模块。

3）增强软件模块服务发现和服务交付协商交互性的模式。

4）移动设备的演进和支持移动中间件的智能设备实例。

5.1　移动中间件技术

现在许多研究工作集中于设计移动环境中的中间件平台。我们讨论4种不同类型的中间件技术（订阅/发布、反射、移动代理和P2P中间件）并介绍一些方案作为例子。每种技术至少支持一个以上的上述3个需求。我们分析了这些中间件应用在移动环境中的优缺点，并将之扩展到资源受限设备。

5.1.1　订阅/发布中间件

订阅/发布中间件，也称为基于事件的中间件，是建立大规模分布式系统的一种模式。发送者和接收者预选无需彼此知道对方。接收者无需指定特定的源，只需订阅感兴趣的内容，而发送者也只需发布相关信息，而无需发给特定的接收者。订阅/发布中间件解释这一订阅信息并向订阅者发送相关的内容。

已有很多标准的订阅/发布中间件规范，如 CORBA NS （通知服务）[4] 和 JMS（Java 消息服务）[5]。

订阅/发布模式支持异步通信和上下文感知。现有的订阅/发布中间件平台已经扩展到支持移动上下文，如基于 Java 开发的使用移动代理的面向对象的中间件。[6]

5.1.2　反射中间件

反射的概念被引入到编程语言设计领域，允许程序接入、质疑和更改自己解释[7]，最近已经应用于分布式系统。反射中间件通过使用元接口提供了系统行为的检查和适配，使得系统开放、可配置以及可重配置。反射中间件是一组协作组件，它能配置小的中间件引擎，以便与其他传统中间件互操作。

动态 TAO 是一个 CORBA ORB 开发的作为 TAO 的扩展[8] 的基于组件的反射中间件平台移动可重配置代理就是建立在动态 TAO[9] 平台上。另一个例子是 XMIDDLE[10]，它使用了移动代码技术进行扩展[11]。

5.1.3　移动代理中间件

移动代码是从一段源端移动到目的端并执行的可执行代码。它可以提高速度、灵活性以及处理断链的能力。为了扩展现有设备和系统的功能，它能够进行动态的代码装载，可解决资源受限问题。

移动代理携带代码和数据进入代码执行环境，如当前状态。它能在无需挂起服务的情况下升级分布式对象。与移动代码不同，移动代理自主导航：它们自主决定是否移动、何时移动。移动代理适合可用带宽有限和网络连接经常断开的移动环境，并支持异步通信。网络连通性要求只发生在代理从一个地方移动到另一个地方时。

另外，根据当前的上下文信息，移动代理可以支持动态适配和个性化。从预定义策略到自学习的人工智能方面，移动代理还能支持一定程度的智能化，并可自动执行。为了实现共同的目标，代理之间还可以相互合作，分担负载。移动代理在动态的、异构的及开放的环境中一定会处理的更有效。

Java 虚拟机和 Java 的类装载机制提供了开发移动代理中间件需要的有用功能。这些功能中最重要的包括串行化、远程方法调用、多线程和反射。目前市场上已经有几个基于 Java 的移动代理中间件平台，它们都与现行标准兼容，包括

OMG的MASIF（Mobile Agent System Interoperability Faeility，移动代理系统互操作能力）或者FIPA（Foundary for Intelligent Physical Agent，智能物理代理基础）。一方面，MASIF定义了基本功能的标准接口和异构移动代理系统间的传输。另一方面，FIPA标准化了代理平台的通用架构，侧重于可互操作的代理通信语言。代理的移动性在FIPA中并非强制要求。IBM的Aglets[12]和SOMA[13]符合MASIF标准，而FIPA-OS[14]，JADE[15]，LEAP[16]符合FIPA标准。

5.1.4 P2P中间件

分布式应用基本有两类通信模型：客户机-服务器模式以及P2P模式。前者是传统的请求-响应方式，应用组件分成两类，一类发起请求（客户机）而另一类完成请求（服务器）。后者是两个应用进行服务发现时交换信息的会话通信模型。在P2P模式中，不再有中心节点发布服务和信息，所有的参与者都可以全局透明地共享信息，它通过P2P的方式实现上下文感知。

P2P中间件具有节点组创建、加入和交互以及发布广告的能力，因此它可以动态地发现需求、适应和变化并在节点间实现协同，以Ad Hoc的方式实现一个共同的目标。它同时能够在多个节点间分担计算负载和共享资源。

动态服务发现是P2P中间件中的关键技术，如JXTA[17]、OMG的SDO（Super Distributed Object，超级分布式对象）[18]以及Apple的Bonjour[19]都是IETF零配置协议[20]的实现。

5.1.5 移动中间件平台的挑战

订阅/发布中间件和反射中间件的方法用于重建传统中间件，以此满足移动中间件的需求。移动代理模式可以作为一个额外的服务应用其中，P2P中间件作为底层通信机制。表5-1列出了各种移动中间件方法的优势。

表5-1 移动中间件技术的比较

	稀少资源	异步通信	上下文感知
订阅/发布中间件		X	X
反射中间件	X		X
移动代理中间件	X	X	
P2P中间件	X		X

反射中间件的可配置性、代码移动性和移动代理中间件的部署以及P2P中间件的分布式计算负载都是支持有限资源设备的。注意上述中间件系统中有一些需要比现在移动设备所能支持的资源更多的资源。设备能力的提升可以解决这一问题。

然而，没有一个中间件能够满足所有的移动中间件的要求。有一些研究将多个平台技术进行整合，如订阅/发布中间件与P2P通信技术的整合[21]或者反射中

间件与移动代理的整合[11,22]。我们预期这些技术的整合在满足移动中间件的需求方面有重要作用。

5.2　结构组件

在本节，我们介绍为支持第 2 章和第 3 章中讨论的下一代网络的新功能和服务架构中间件平台所要求的功能组件。图 5-2 是中间件的逻辑结构。

网络支撑层提供了异构网络中的网络通信控制功能，在 2.2 节已经讨论。用户支撑层较传统的服务中间件增加了自治和主动代理，这使得基于上下文感知、个性化和适应性的以用户为中心的服务得以提供，这部分内容在 3.2 节已经讨论。接下来我们讨论服务支撑层的各个功能模块。

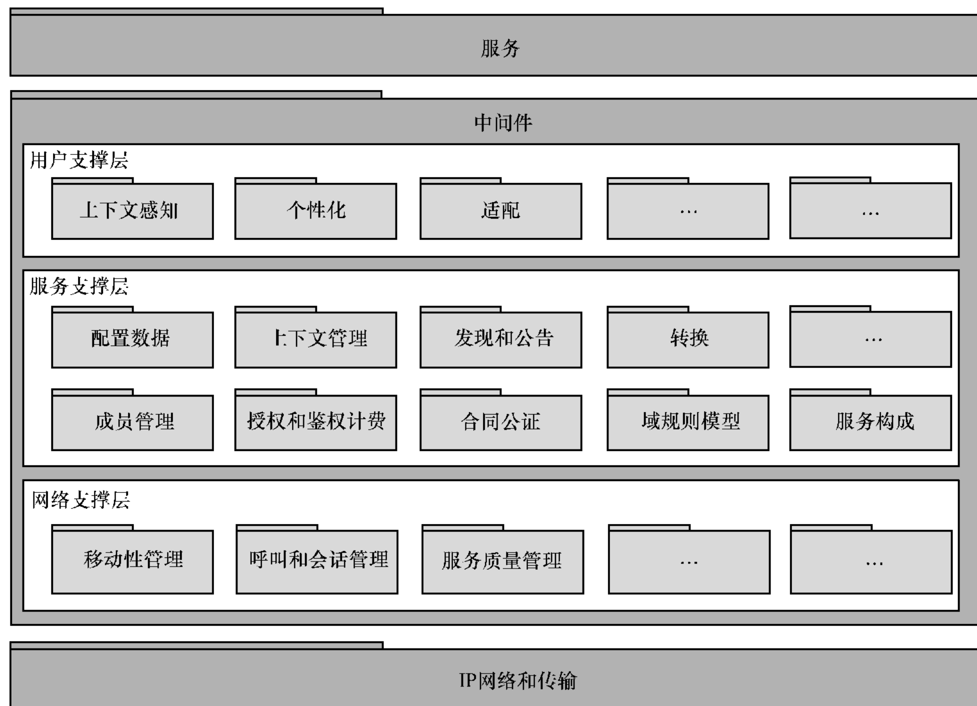


图 5-2　中间件逻辑结构

5.2.1　服务支撑层

1. 配置文件

配置文件指的是适配用户在特定环境中或个人偏好的用来配置某种服务的数据集合，例如个性化服务配置等。配置文件可以分为以下几类：用户配置、服务

配置、网络配置和终端配置。这一中间件功能模块通过向用户提供选择、查询、存储和更新配置的方法支持对配置的管理。用户可以有多个配置并选择最合适的配置，配置也可以基于其他标准自动选择（更多细节参见第7章）。

2. 上下文管理

上下文是一组用来决定应用行为或事件发生的环境状态。上下文可分为以下4类：用户上下文（如用户的位置和当前的社会状态）；时间上下文；物理上下文（如亮度、交通情况、温度）和计算上下文（如通信速率、可用内存和处理能力、电池状态）。这一组件提供了向服务提供上下文信息的方法。该组件包含感知和查询当前上下文（如通过传感器）以及将其传递给其他服务的机制。它同时记录一个时间段内的上下文历史记录。此外，它还负责监控上下文的变化并通过订阅/发布接口传递给其他组件（更多细节参见第9章）。

3. 发现和公告

发现和公告用于动态发布和搜索描述服务或资源的信息，它同时提供了注册服务和注册时注册、更新和删除动态需求资源的方法。

4. 转换

转换支持服务处理和产生的内容适配。有几个方面可以转换：格式和编码（如GIF和JPEG文件）；类型（如视频、音频和文字）；以及结构（如XML和HTML）。

5. 成员管理

成员管理提供了管理组和成员的方法。组可以用来方便地指定和表达有共同兴趣和目标的实体。它可以帮助管理组信息的动态变化，如增加成员、删除成员和管理成员的生命周期。

6. 授权、鉴权和计费

AAA用来识别和鉴权，通过允许基于一系列授权策略规则的接入控制来确认和保证授权。计费的目的是处理和管理帐户相关信息，计算用户使用相关服务及资源、应支付的费用。

7. 合同公证

合同公证用于管理和确认合同有效。合同持久存储于一个仓库。合同公证也支持库存合同的搜索、注册、删除、更新和确认。

8. 域规则模型

域规则模型接受给定的一组结构化数据，以规则或本体形式返回形式化模型，该模型由域专家创建。规则或本体可以是简单的弱语义关系的简单分类，也可以是包含更复杂的连接语义和规范的概念模型。域专家使用适合模型化连接和关联定义的规范语言在组件中存放域模型（更多细节参见第7章）。

9. 服务构成

服务构成的目的是创建和从一系列服务组建中动态提供新服务。它实现现有

服务的集成，用来满足多个服务合作时的复杂请求。

5.3　动态服务交付模式

动态服务交付模式是 5.2 节讨论的中间件功能模块互联时的配置实例。它支持异构无中心节点环境下实现服务合约和交付的服务发现和自治协商，其中服务将被注册或发现。实体可以是任何的软件或软件组件。他们可以有多重身份：像参与者、协调者（在协商过程中）、用户以及服务提供者（在服务交付过程中）。

动态服务交付模式包括以下 3 个阶段：

1）介绍阶段：实体之间通过订阅/发布接口互相交互，实现服务发现和广播。

2）协商阶段：与第一阶段相同的实体充当参与者，一个信任的实体也参与进来充当协调者。为了实现服务合约，根据协调者批准的计划，参与者之间交换协商计划、信任文书和提案。

3）服务交付阶段：充当用户和服务提供者的实体彼此交互，完成服务合约规定的款项。服务会话管理者管理服务交互并在完成服务交付时终止服务会话。

图 5-3 描述了服务实体以及它们的作用。实体 A 和 B 对某一服务有共同的兴趣，分别充当用户和服务提供者。在协商过程中两者均充当参与者。实体 B 还另外充当协调者的角色，根据协商规则处理参与者之间的交互过程。最终在服务交付阶段实体 B 向 A 提供服务。

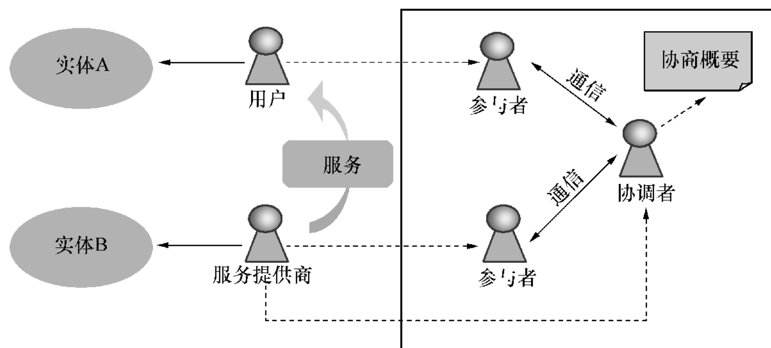


图 5-3　动态服务交付模型中实体角色举例

图 5-4 显示了根据图 5-3 中的角色分配进行动态服务交付模式的顺序。协商开始于确立协调者。是否同意协商规则是可选的。协商规则定义了协调者应如何展开协商过程的规则和协议。协调者可以预定义协商规则。协调者邀请参与者递交原始提案，如基于参与者的协商策略的价格或需求。协调者检查参与者的需求是否吻合；如果不吻合，协调者根据协商规则建立更新的提案并返回给参与者。参与者根据协商策略确定更新的提案是否能够接受。一旦达成服务合约，协调者

建立服务合同并确立一个服务会话管理者控制服务会话。服务会话管理者监控参与主体的交易保证服务交付。

该模式用于实体需要与其他实体交互时，通过支持系统内和系统间的灵活交互使系统支持中间件以及应用/服务层的重配置和适应性。

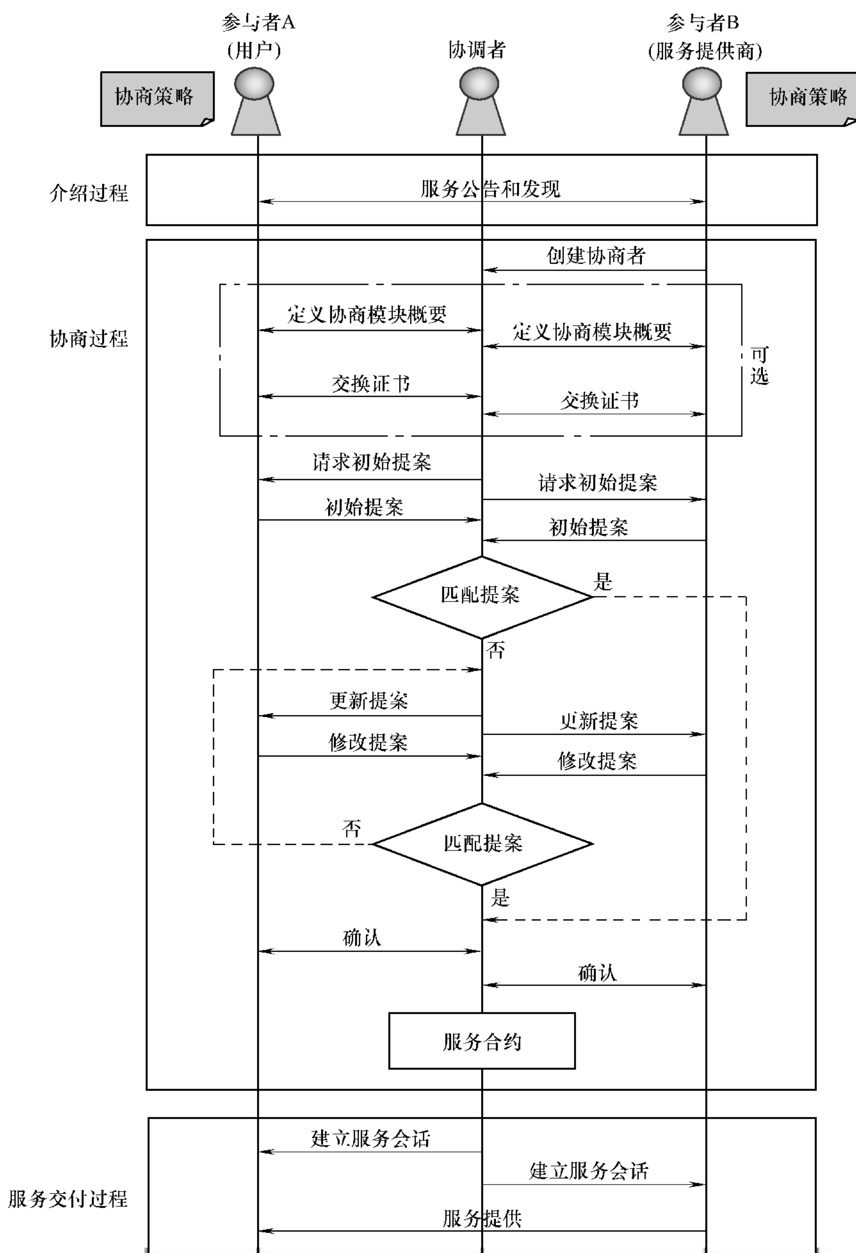


图5-4　动态服务交付模型顺序

5.4　智能设备支持移动中间件

在过去的 5 年中，我们看到了移动设备演进的两个主要方向。其一，高性能移动设备，主要用于人与人或人与机器通信，它们大多支持网络浏览器、照相机以及音频/视频接口。其中一些也将智能卡技术集成进来，支持信用卡功能。我们称之为“一体”设备，它带我们走进一个只需移动设备的环境，这与现在高端移动设备的演进一脉相承。其二，以机器为中心的通信中的“全设备”，如现在市场上专门针对数据服务的移动终端，它们支持额外的射频接口，如 GPRS、WLAN、Wi-Fi 和 WiMAX，这些接口可以连接到 PC 中。在将来，硬件设备的演进将能让“全设备”最小化到一个芯片上，即所谓的“全芯片”。它们将可以嵌入或与许多类型的设备相连接，实现与用户环境的积极交互。我们预期，“全芯片”将成为泛在时代的主流，而“一体化”设备也将与之共存，作为主要的个人移动设备（见图 5-5）。

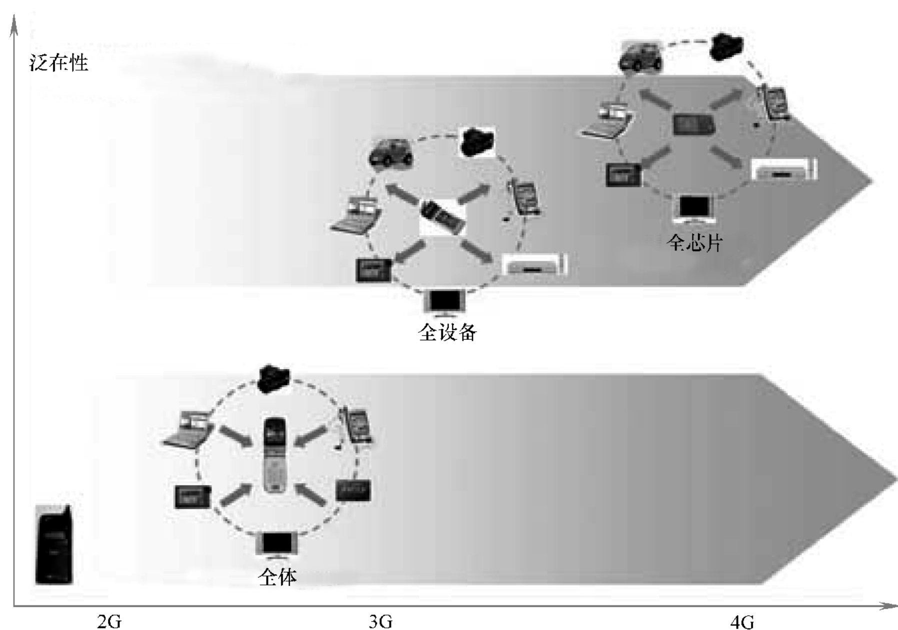


图 5-5　移动设备的演进

除了移动设备，还有另外一种类型的用户手持设备。智能卡，即所谓的 SIM 卡，作为私人的安全设备在 GSM 时代广泛使用，最近也用于 UMTS 移动系统。通过存储用户身份和支持网络鉴权，它们在运营商和用户之间建立了直接的联

系。考虑到移动设备的演进场景，我们预言下一代的智能卡作为用户和服务提供商的交汇点将起到重要作用。与大多数用户不可见、集成的泛在设备不同，智能设备是个人的可见设备，提供了安全的应用操作环境和私人敏感信息的存储空间。

在下文中，我们将考察智能卡相关技术的发展现状，并讨论如下几个趋势：

1）移动通信中的智能卡，如SIM/UICC卡存储用户信息和未来支持开放执行环境的能力。

2）作为迈向泛在的一步，无签约智能卡将被嵌入移动电话。

3）移动设备通过RFID支持读入功能。

4）智能设备的演进，在泛在环境中与智能卡起着相似的作用。

5.4.1 智能卡技术

智能卡，即所谓的SIM（用户识别模块）和3G中的UICC卡（通用集成电路卡），今天在GSM和UMTS网络中已经广泛使用。

1. 物理结构

智能卡，通常称为集成电路（IC）卡，是嵌在一块塑料卡上的集成电路芯片。它们包含数据传输模块、内存和存储模块以及处理模块。

智能卡中的关键模块是一块包含通信接口的防水集成电路芯片。在20世纪90年代，标配8位CPU，而在现阶段，为适应高性能的需求，如支持Java虚拟机，通常需要时钟频率为30～50MHz的32位CPU。现在多数芯片也装配专用协处理器用于加密功能，这比使用软件处理具有更高的计算性能。

智能卡中既有非永久性存储器也包含永久性存储器。对于快速读写，现在市场上有一些比Flash更便宜的技术，如FeRAM（铁电随机访问存储器）以及MRAM（磁性随机访问存储器）。在20世纪90年代早期，芯片的内存通常小于10KB，而今天市场上的产品已达到1MB，应用于多媒体中。

2. 软件结构

智能卡的软件结构基于文件系统以及相关的访问命令集（ISO7816-4标准制定）。安全命令如鉴权或个人信息验证通常在企业标准中定制和扩展，如用于通信的SIM/USIM中的ETSI的SCP或3GPP以及用于支付系统的EMV[25]。

智能卡的操作系统提供了基本功能，如通信、存储控制和加密等。智能卡操作系统最重要的作用是控制智能卡硬件的电子和物理接口。虚拟机提供了允许多应用下载和运行的环境。操作系统和虚拟机也有企业标准，如Java Card[26]以及MULTOS[27]。MULTOS不仅可用于虚拟机，还可以用于应用验证过程以及其他操作过程，因为它最初是为在智能卡上加载信用程序所设计（如万事达卡）。Java Card是最广泛使用的独立应用平台。SUN微电子公司定义了Java标准的最小集

合 Java Card。Java Card 定义了与 Java 兼容的微型虚拟机和运行环境以及智能卡专用 API。应用开发者可以开发 Java 程序（Java Applet），包含文件结构和命令集。图 5-6 显示了 Java Card 的结构。

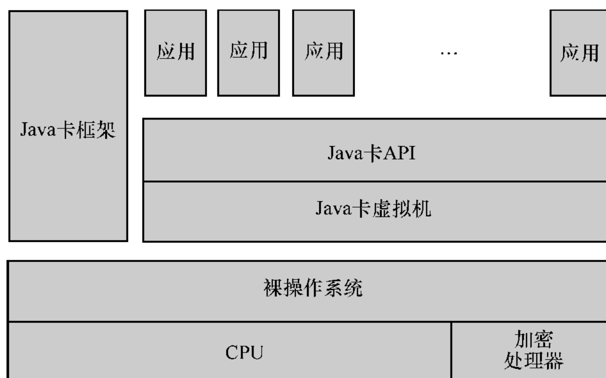


图 5-6　Java Card 的结构

3GPP 和 ETSI 的 SCP 定义了 SIM/USIM 使用的应用工具包（SAT/USAT），这样，可以主动使用它们而不致使其作为外部读入程序的从动程序。SAT/USAT API 的设计便于在空口或蓝牙端口开发和下载 SAT/USAT 应用。SAT/USAT 能将应用存储于 SIM/UICC 上，以便与移动终端交互和操作，操作包括显示文字、发送短信息以及通过 URL 进行上网等。

5.4.2　无签约智能卡技术

另一种类型的智能卡通信机制是无签约，它无需金属接口。这种技术在公共交通领域很普遍，如支付交通费用和存储电子客票（香港的 Octopus 卡[20]，伦敦的 Oyster 卡[29]以及巴黎的 Navigo 卡[30]）。无签约智能卡通过无线电产生电磁场使用外部读卡器供电。ISO/IEC 10536、ISO/IEC 14443、ISO/IEC 15693 根据距离远近定义了不同类型的通信协议。ISO/IEC 10536 被称为近耦合，意味着允许的通信距离非常短，卡需要使用 4.91MHz 的频率与终端相连。遵循 ISO/IEC 14443 标准的智能卡称为近距离卡，允许通信范围从 0.2mm ~ 10cm，通信频率为 13.56MHz。ISO/IEC 15693 卡成为邻近卡，允许通信范围为 70cm，通信频率为 13.56MHz。

在移动通信系统中广泛使用的无签约智能卡的例子是 NTT DoCoMo 在 2004 年 6 月推出的 i-mode FeliCa“移动钱包”服务。在 2007 年，超过 2000 万终端配备了 FeliCa，超过 2.2 万个商店支持该服务，包括电子货币和电子票。FeliCa 是索尼公司开发的一种无签约智能卡技术[32]。无签约接口通信协议兼容 ISO/IEC

18092 标准，该标准作为近距离通信（NFC）标准由索尼和菲利普公司提出。NFC 支持 13.5MHz 时钟频率下的 212kbit/s 的通信速率。所有的处理过程，包括卡检测、多方认证到数据读写，均在 0.1s 内完成。FeliCa 支持读/写命令以及电子商务应用中的特殊命令。目前可用的 FeliCa 卡支持 8 位 RISC CPU 和 2～4KB 的用户/应用存储。

在 i-mode FeliCa 中，FeliCa 芯片和与外界进行非签约通信时使用的环状天线集成于移动设备中。FeliCa 芯片可通过外部设备的读/写器的微弱电信号进行操作。FeliCa 应用进一步与移动设备上基于 J2ME CLDC（有限连接设备配置）和 KVM（K 虚拟机）的 i-appli 进行集成[33]，这可以让用户下载和更新（如充值）在线 FeliCa 应用，以及本地访问存储于 FeliCa 上的数据（如移动电话作为数据的阅读器）。

i-mode FeliCa 用于电子商务、电子货币、电子票以及身份存储，如雇员状态和会员信息。在 2006 年东日本铁路公司（JREast）宣布了一项基于 i-mode FeliCa 的服务，可以使用手机作为客票和充值工具。这一现象可能引起以机器为中心的通信模式的增长以及与环境的交互，通过"一体化"设备的形式，作为迈向泛在性的一步。

5.4.3 RFID 技术

RFID（射频识别）技术在射频状态综合使用电磁学和静电学技术，通过存储的数据来惟一识别一个物体、动物或人。RFID 也叫做射频标签或智能标签，用在物流、配送和供应链应用中[34]。与条形码不同，RFID 在制作和配送环节不怕潮湿和高温，无需实体接触就能实现自动识别。读入范围根据 RFID 的种类可以从几英寸到几百英尺，不过这需要使用不同的频率，ISO/IEC 18000 对此进行了规范。这一标准规范了射频接口和协议，但未规定 RFID 的物理格式。因此，RFID 在形状和尺寸上大小各异，有塑料卡、粘标签、袖口标、硬币、标签等。RFID 可以内部供电，从而进一步提高灵活性，这也与智能卡有所区别。

RFID 仅在外部读卡器通信范围内进行通信，生命周期长，其经久性是显而易见的。RFID 标签主要有两类：

1）主动射频标签既可通过内部电池供电，也可通过外部读卡器供电。主动射频标签比被动射频标签更贵，也更大。但是，其功能更强大，读入距离更长。

2）被动射频标签通过外部读卡器产生的场供电。被动射频标签通常更轻、更便宜，可以无限次使用。但是其读入距离更短，与主动射频标签相比，需要更大功率的读卡器。

为了降低成本，现在多数 RFID 使用内存卡，不具有计算能力。外部设备仅简单地读出存储于 RFID 的数据。我们可以认为可以增强 RFID 支持更多的功能，

如智能卡中的计算和防篡改。这样无签约智能卡和 RFID 的边界变得越来越模糊。目前，市场上已经出现充当 RFID 读卡器和支持 NFC 接口的移动终端[35]。

5.4.4　智能设备举例

一些新的应用场景正在涌现，如游牧用户装备智能设备，这些一方面通过泛在性设备与用户环境建立起持续的交互，另一方面也要求更高程度的上下文感知和软件对变化的适应性。在不久的将来，智能设备将像现在智能卡在移动系统中一样，扮演相同的角色，提供网络安全的身份，建立运营商和用户之间的交互点。我们期待短距离通信，如 NFC、RFID 或 ZigBee 进一步融合，成为传统移动网络和传感器网络之间的协调员。这将使智能设备融入泛在环境中。智能设备可以和与日常物体相连的泛在性设备进行交互，也可以组成传感器网络，支持更大距离的应用和服务。它们也可以充当本地 Ad Hoc 网络的网关，在泛在环境中提供增值服务。

为了支持上下文感知和软件对变化的适应性，智能设备将整合进中间件平台。随着设备硬件的演进和开放软件环境的发展，它们可能成为分布式的目标。但是，与其他实体做比较时，如移动“一体化”设备、外围设备（家庭用具）、应用服务器和网关，它们仍只有有限的资源和能力。智能设备只有基本的软件功能。如果不考虑硬件或通信层，智能设备与中间件的适配将加速服务的发展。比如，智能设备可以成为用户个人软件代理最合适的归属。这样用户永远可以携带其个人代理，且可以一直到其他地方，在本地或广域网中自主但安全地执行任务（如与其他代理协调）。

这种方法可以通过扩展移动代理中间件，使之支持 JXTA[36]来实现。在智能设备上装载代理软件模块的一个好处是它可以在本地和 Ad Hoc 网络中使用，但其中的节点不能时时可达。另一好处是其安全性和私密性，如在本地保护用户敏感信息，利用加密功能[37]建立代理联盟。

另一种方法由 Blefari-Melazzi 等[38]提出的。智能设备已经用作“简单设备”，在异构的移动和固定网络中支持用户网络接入和服务使用。简单系统包含 3 个主要模块：智能设备、终端代理和网络代理。智能设备可以插在不同的终端中，它存储用户信息、偏好以及策略。终端代理和网络代理基于分布式代理结构，支持发现、广播、适配和呈现。智能设备方便个性化机制的实施，探索服务实现、驱动终端能力自动适配以及服务适配到不同的网络技术和能力。

5.5　小结

最近移动通信中涌现的泛在网络和服务带来了运行于异构设备中的新的中间

件解决方案的需求。这一需求已得到确认，移动中间件正成为一种解决移动、泛在和资源受限设备的挑战的特殊中间件。本章讨论了移动中间件领域现在的一些工作，包括扩展传统平台支持移动和资源受限、建立新的功能模块以及改进软件模块更适合服务发现和服务交互协商。我们也讨论了移动设备的演进，包括智能卡，并给出了一些支持移动中间件的智能设备实例。

第 6 章　跨层设计 —— 一种新的移动通信系统优化方法

Marco Sgroi，Wolfgang Kellerer

跨层设计（Cross Layer Design，CLD）是一种允许跨越传统层界限优化通信网络架构、但并不改变现有通信网层架构的新方法。在本章中我们将阐述 CLD 的基本原理以及如何应用 CLD 来设计和优化一个无线视频流媒体系统。

6.1　简介

下一代无线网络需要支持一些复杂且耗资源的应用，如视频会议、3D 导航和交互式游戏等。网络运营商将面临如何有效分配无线媒质资源以提高网络容量和为最大规模的潜在用户提供最高质量服务的挑战。无线资源分配的问题很难解决，这主要是因为无线信道的时变传输特性和大多数应用 QoS 的动态变化需求。在最差情况下，静态配置网络将导致较差的性能和对资源的低效率利用。相反，网络应该能够动态地调整配置来适应环境的变化，上下文应用和无线信道条件都属于环境范畴。动态调整需要在各层之间及时地交换信息和对网络运营期间协议层参数的周期性重配置。

网络架构的传统设计遵从分层原则。层就是一组在同一抽象级别上运行的通信功能，比如同样的运行速率或者处理相同大小的包。因此，一个网络的通信功能可以由几个层来构成，每一层都对上层提供服务，并且使用下层提供的服务。OSI 参考模型就是一个常见的分层思想的例子，如图 6-1 所示。

分层的思想常常被用来设计网络架构，这是因为：

1）它应用了分离的基本思想，并且简化了设计任务。

2）它有利于模块化，并且允许替换单独的一层而不改变整个协议栈。

然而，纯粹分层并不能完全满足设计下一代移动系统的需要。研究者们已经指出，在某些场景下，严格分层的架构在性能上要差于多层可以联合优化的架构。这些场景主要可以分为两类：

1）各层在本地优化并考虑不同指标。各协议层以不同的指标进行本地优化有可能导致网络整体的不可预期性。Kawadia 和 Kumar 在参考文献［2］中描述

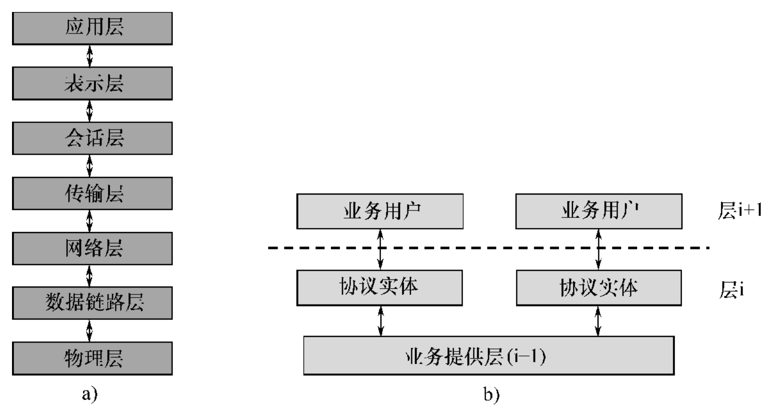


图 6-1
a) OSI参考模型 b) 分层架构

了协议层之间作用冲突的一个例子：网络层为了优化延迟使跳数最少，而MAC层根据接收到的信号强度改变其数据传输速率。在网络层选择跳数少的路由将导致链路的数据传输速率降低，并导致整个吞吐量较低。原因是网络层和MAC层各自根据不同的指标参数（网络层使用延迟参数，而MAC层使用吞吐量）进行优化。网络层没有考虑上层服务的吞吐量需求，并选择了一个最小延迟路由，而没有考虑链路的吞吐量能力。

2）各层没有考虑其他层产生的报文的语义。提供高的QoS要求各层能够理解接收的其他层的包的语义。例如应该能够确定来自上层包的优先级，并能在网络拥塞的情况下丢弃最低优先级的包。理解来自下层包的语义能够让上层更好的适应物理媒介的动态变化。例如，TCP通常假定底层媒介是可靠的，相对于缓冲拥塞导致的丢包，噪声导致的丢包基本上可以忽略不计。这个假定对于无线媒介来说不是总有效。因此，当TCP运行在无线信道上时，由于高噪声和干扰导致的丢包会被TCP误解，并降低传输速率来避免网络拥塞，结果导致在无线链路上TCP的吞吐量相当地低。在这种情况下，问题不是出在分层方法本身，而是因为TCP无法区分出不同类型的丢包。为了解决这个问题，参考文献［3］提出了显示拥塞通告机制。

流媒体视频中提供了另外一个协议层不理解报文语义的例子。通常一个视频流由一些视频帧序列构成，这些帧在解码时相互之间部分依赖，因此在容错显示方面具有不同的重要性。目前，无线传输系统的资源调度并没有考虑服务参数，如独立和依赖帧的不同重要性以及当丢帧时导致的失真。我们使用这个例子来阐述6.4节和6.5节的无线网络中跨层设计的可能性和基本原理。

为了克服纯粹分层架构的缺点，有研究者最近提出了一种新的跨层设计

（CLD）方法。CLD 考虑层之间的相互依赖性和交互，并且允许跨层进行优化。关于 CLD 常见的误解是认为网络是不分层的。分层只是一个用来简化网络设计和管理任务的假象。在具有抽象、可确定层的架构中，CLD 允许多层参数的联合优化。因此，CLD 应该被看作是分层方法的一种替代方案，而不是一种补充。分层和跨层优化可以一起使用来设计适应性强的无线网络的工具。

Clark 和 Tennenhouse 最早指出由于太严格地应用分层原理，可能会导致效率丧失[4]。他们提出采用应用级别的组帧和集成层的处理技术来优化跨层的协议栈实现。

最近，CLD 被主要应用在功能级别，用以联合优化多层的参数。Hass 指出，分层架构模型一个最主要的缺点是缺少各层之间共享的信息，这将使得网络无法快速地适应环境的变化[5]。Shakkottai 等人强调从底层到协议栈的最高层需要考虑由于用户移动导致的无线媒介的时变特性的重要性[3]。如果多个用户共享同一个物理媒介，多用户的多样性增益将可以通过给那些传输成功概率高的用户分配信道资源的方式获得。如图 6-2 所示，一个系统由几个移动用户和一个基站构成。基站调度器基于每个用户的信道状态（例如根据传输和差错率）动态地给用户分配时隙。

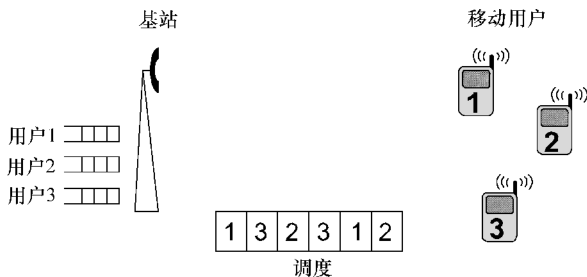


图 6-2　多用户信道调度

Kawadia 和 Kumar 对 CLD 持有不同的观点[2]。首先，他们强调模块化和重配置的重要性，分层架构可以更好地支持它们。其次，他们指出大部分 CLD 的文章都没有考虑当出现了多个跨层交互和网络参数受冲突适应环影响时跨层优化的全局效果。Rappaport 等人指出了跨层设计和自适应性之间的区别[6]。前者被定义成跨越传统层界限的一种静态优化，而后者则被定义成当无线媒介变化或者服务应用需求时动态发生的一种优化。Adve 等人提出了一个各层动态地适应和协作以实现全局优化配置的框架[7]。资源的分配由一个拥有全局信息的中心资源管理器控制。资源管理器检查服务对整个系统上下文状态的需求，并且选择能最大化整个系统效用和满足资源限制的配置。只要没有超出分配的资源，在几个特定层内就有可能进行本地的调整。对于本章，我们定义 CLD 为一个协议栈几个

不同层的动态调整用来优化某个全局的目标，如用户感知的服务质量。这种优化基于6.2节中描述的跨层功能架构。

目前CLD技术的应用主要是用来优化无线上的TCP[8]、蜂窝网络信道调度[9]、自组织网络和传感器网络的协议设计[10,11]和无线视频流媒体[12-15]。

Ludwig提出一种自上而下的方法用来提高无线上的TCP性能[8]，在这种方法中，对QoS的需求从传输层一直延伸到数据链路层。Sternad使用信道预测在时间（时隙）和频率（OFDM载波）上对无线媒介资源进行调度[9]。Xylomenos和Polyzos定义了一种新的链路层用来提供区分服务从而满足服务多样性的需求[10]。

在传感器网络中也非常有可能应用CLD技术[17]。传感器通常深埋于地理环境中，因此网络很容易受到外界变化的影响，如能量损耗、节点移动性和变化的干扰等，而且需要传感器网络能快速地适应这些外界变化。在传感器网络领域，CLD主要被用来做联合优化：

1）网络和MAC层，用来组合路由选择和节点调度策略。

2）中间件和网络层，用来根据节点连接的变化调整中间件。

6.4节中将详细讨论如何应用CLD到视频流媒体中。

尽管CLD技术在某些场景下已经被成功应用于协议栈的设计中，但是仍然需要进一步的研究来解决一些基本的问题。在下面的章节中，我们列举了CLD的一些基本原理和一些基本问题，作为6.5节中的一个范例：

1）当考虑性能增益和实现代价的时候，什么时候应该应用跨层优化（参见6.3节）?

2）跨层优化应该如何实现？作为一个中间单元还是作为一个分布在多网络节点和层之间的一组元素（参见6.3节）?

3）优化器使用哪个参数抽象可以精确并易管理的描述协议层的状态？为了能实时地计算系统层的最佳设置，通常只能考虑所有系统层可利用特性中的部分参数（参见6.2节）。此外，为了能够更容易地输入进优化器，参数必须预先计算（抽象）。

4）什么标准应该用来优化参数a，尤其是当它支持多个不同类型的并发服务时？针对最差性能用户感知的最佳用户服务质量可能是一个用来优化系统的标准，而在给定某些服务质量阈值时，为全部用户感知的服务质量就有可能成为另外一个标准。

6.2 跨层功能架构

一个跨层架构（Cross Layer Architecture，CLA）包括多个协议层和一个跨层优化器（Cross Layer Optimizer，CLO）。跨层优化器对网络的多个协议层进行联

合优化，获取它们可预测状态的抽象，并且找到它们参数的最佳值。图 6-3 给出了一个跨层架构。

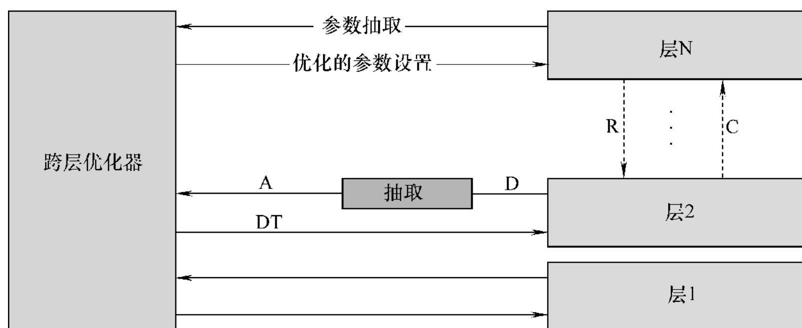


图 6-3　跨层架构

跨层优化主要包括以下 3 个步骤：

1）层抽象，用来计算协议层可预测状态的抽象。

2）优化，用来找到优化一个特定目标功能的层参数的值。

3）层重配置，用来将参数的最佳值分布到对应的层中。

这些步骤按一定的速率周期性地重复，该速率的设置基于服务需求变化的快慢和物理媒介的传输能力。

确定一套适合描述协议层状态的参数十分重要。一个由很多参数组成的描述可能是准确的，但通常导致较高的额外通信开销（由于在优化前需要从网络中搜集所有的参数）和额外计算开销（由于在优化时需要探测参数空间）。因此，要使用抽象来减少 CLO 需要的参数的数量。选择用来联合优化的层参数也是一个重要的步骤。虽然一些层参数可以直接由 CLO 设置，但其他一些参数却无法直接设置，它们在其他参数设置好后可以确定。协议层参数可以划分为以下几类：

1）可直接调整（DT）的参数。这些参数可以直接由 CLO 设置。比如，TDMA 系统中的时隙分配或 OFDM 系统中的载波分配。

2）可间接调整（IT）的参数。这些参数无法由 CLO 直接设置，但是在 DT 参数设置好后可以确定。比如，依赖于编码类型和调制方案的数据传输速率。

3）可描述（D）的参数。这些参数可以由 CLO 读出，但是不能调整。比如，流媒体视频服务中，在编码时设置的帧率或图像尺寸。

4）抽象（A）参数。这些参数是可描述参数的抽象。比如，Peng 等人在参考文献［12］中描述的帧丢失概率，就是来源于 Gilbert-Elliott 模型中的信道状态传输概率。

层参数也可以根据另外一个维度：网络层之间的交互来划分。

1）能力（C）参数定义网络层提供一个服务的一些属性。

2）需求（R）参数定义网络层要求有的一些属性。

在一个分层架构中，各层对其上层提供服务，这些服务的质量根据该层的一些参数来表示。一个参数可以是C参数或R参数，取决于该参数是从提供服务的层还是使用该参数的层的角度来考虑。比如，对一个需要与底层通信延迟最大的层而言，参数延迟就是一个必要条件。而对底层而言，延迟是一个描述其能力的一个参数。当跨层优化不是作为一个中间单元实现，而是由位于不同层上的一套本地优化器实现时这个微妙的差别特别重要。必须正确地匹配各层的需求和能力，以确保本地优化器能相互配合，从而实现网络的全局优化目标。

6.3 跨层优化的实现

和纯分层架构相比，跨层优化提高了网络性能和适应性，但有可能也引入了额外的实现成本。主要包括3种类型的成本：

1）计算成本。CLO需较高的计算能力来确定一系列参数的值；评估一个复杂的目标函数时也需要较高的计算能力，并可能引入相对较大的处理延时。参数抽象有助于降低复杂性，但却有可能降低生成配置的最优性。另一个降低计算成本的方法是使优化器成为一组同时运行但可能在不同资源上执行的组件。

2）通信成本。CLO使用在分布式网络位置上可用的网络参数。收集这些参数会导致较大的带宽额外开销。

3）重配置和管理成本。分层架构由一组协议层构成，每层单独定义，并且通过良好定义的接口可以和其他层区分开。跨层架构模块化相对较差，因此当有变化时更难管理和重配置。这种类型的成本不太容易量化，然而，它可以通过定义传统层和跨层优化器之间的接口的方式来加以限制。

选择跨层架构的一种有效实现需要对性能增益和上述的成本因素做一个仔细的评估。

CLA的实现可以是集中式的，也可以是分布式的。

1）集中式。CLO作为一个集中的单元，从网络层收集所有相关的参数，执行优化，然后将选择的参数值分配给相应的各层（见图6-4）。由于一些原因，集中式方式实现起来通常成本较高，并且效率低。首先，从分散的各地收集网络参数耗时，而且延缓优化过程。其次，层参数以不同的速率变化（物理层的变化量级是毫秒，而应用层变化的量级是秒），因此在最差情况优化所有的参数效率可能会相当低。第三，同时给大量参数计算目标函数也许成本太高。

2）分布式。CLO由一组分布在网络各层（垂直分布）或节点（水平分布）中的组件构成。每个组件执行一个针对全局优化问题参数子集的本地优化，并和

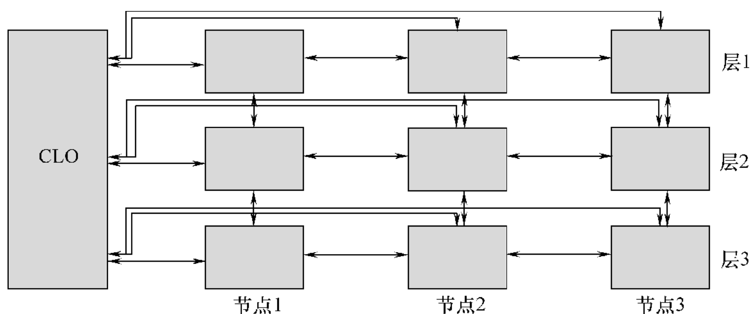


图 6-4　集中式 CLO 实现

其他组件相互合作，以实现全局网络优化的目标。如图 6-5 中所示的采用分布式实现的优化器，其组件属于多个层和节点。垂直上分布的实现有一个分层结构，在该结构中 CLO 放置在不同层上的组件以不同的速率操作，并且使用较低层能力和上层需求的抽象表示来优化本地参数。结果，一个垂直上分布的 CLO 实现和一个分层架构很相像。

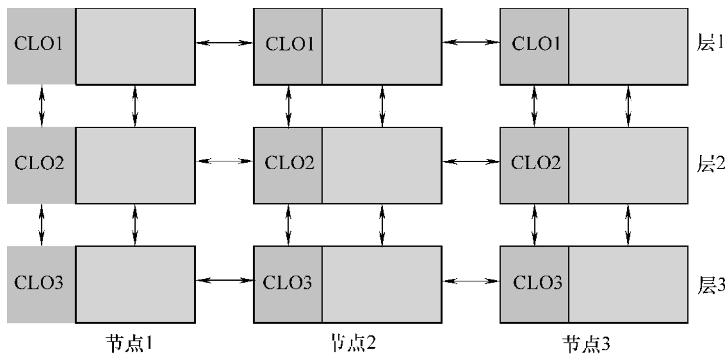


图 6-5　分布式 CLO 的实现

分布式实现通常更为实际和有效，特别是当优化需要大量参数时。每个分布在各层或节点上的 CLO 组件优化一组网络参数。设计一个分布式 CLO 的最大挑战在于如何确保所有本地优化器通过一个良好定义的接口交换一组参数，并且有效地合作来达到全局优化目标。CLO 分布于各层的架构中，每层包括一个本地优化器，该优化器通过考虑上层的需求和下层能力来选择层参数的值。因此，需求必须从应用层由上至下传递，而以一组可行参数值形式（如差错率、延迟和吞吐量等）表示的能力则必须从下层向上传递。

图 6-6 给出了多用户视频流媒体场景下，为了做信道调度，跨层优化的组件垂直地分布在两层中。上层的优化在每个图像组（GOP）的开始重复地执行。

上层的优化器选择分配给每个用户的时隙，并且基于下层在下一个GOP周期(长期地信道预测）时关于可利用信道速率的信息的基础上选择视频源速率。选择的时隙数然后作为需求传递给下层的优化器用来分配特定的时隙和载波，并且选择调制方案。底层的优化基于短期的信道预测，因此执行速率较高。

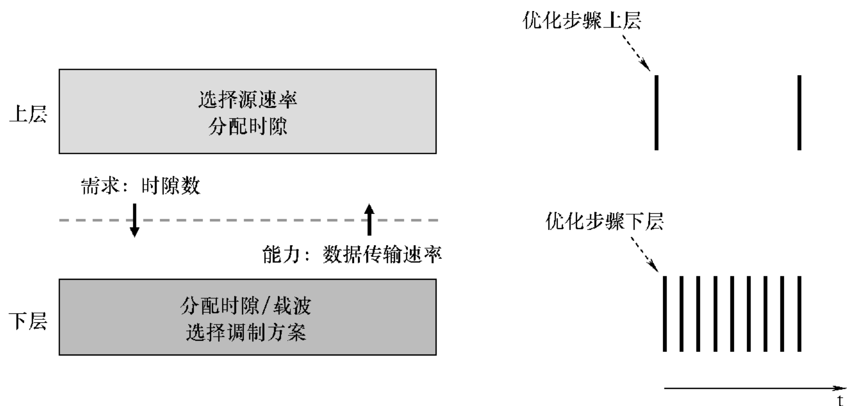


图6-6 多层优化：信道调度

6.4 视频流媒体系统中的跨层优化

在无线媒介中传送高质量的视频要求网络有足够大的带宽，并且能够适应应用层需求和物理媒介能力的动态变化。

视频流媒体应用的需求是随着时间而变化的。由于捕获场景的动态属性和采用的编码策略，视频数据传输速率会有所不同。视频流通常编码成一个连续帧的组序列，称为“图片组（GOP)”。在GOP中，第一帧是I帧，其编码独立于其他帧，剩余的帧（P帧和B帧）根据同一个GOP里的其他帧采用不同的编码方式。只有当正确接收和解码GOP中所依赖的所有帧都被正确接收和解码后，在接收端才能够成功地解码。当丢失帧不同时，接收端的失真也不同。比如，I帧丢失时导致的失真要高于P帧或B帧丢失时的失真。因此，为了优化无线视频流媒体的传输，网络在分配资源时应该考虑服务参数，如包的相对重要性和当丢包时导致的失真。同时，应用层应该适应信道的时变特性，如动态选择能最好匹配当前传输能力的视频服务器的传输速率。

跨层优化能够帮助无线视频流媒体系统快速有效地适应变化。它可以通过其他层的参数来优化单层。比如，如果能够预测针对所有用户的信道的状态，并且了解在任何时刻每个包携带的帧的类型，那么最重要的帧就能被调度在传输能力最强的信道上传输。当用来联合优化多层参数时，跨层优化甚至更为有效。比如，当分

配信道资源时选择服务器视频流的速率，而不是仅仅认为速率是个指定的参数，这样给优化增加了另外一个自由度。有关论述请参见 Khan 等人的文献［18］。

以前在无线视频流媒体系统中应用 CLD 技术的研究大多数集中在使用应用层或物理层的信息来优化单层的参数。Krunz 和 Tripathi[19] 提出通过同步多视频流的相位来分配信道带宽，从而使得峰值速率周期在时间上不会重叠（复用增益）。Tupelly 等人[15] 为多视频流定义了一种机会调度算法。该算法使用一个依赖于信道条件、帧的重要性、队列大小和复用增益的优先级函数。Gross 等人[13] 针对在 OFDM 信道上传送 MPEG-4 视频，提出了 3 种机制用来获得性能增益。第一种机制称为语义序列管理，该机制使用包的相对重要性（MPEG-4 中的 I 帧，P 帧和 B 帧）来决定最终丢弃的帧。第二种机制称为资源分配，该机制在存储准备发送给移动终端的包的基站中，基于队列的长度分配 OFDM 子载波。第三种机制给每个移动用户指定子载波。蜂窝能力和用户感知的质量被用来作为性能指标。Zhang 等人[14] 提出一种在无线上传送多媒体的框架，该框架基于结合了应用层、传输层和链路层的跨层架构。这个架构包括一个服务器、一个基站和移动终端。针对每个协议层，根据使用的其他层的参数，讨论了一些功能，如网络条件估计、网络自适应的非平等保护、应用自适应的 ARQ 和基于优先级的调度等。网络自适应的非平等保护使用应用层的信息，将媒体分成两类（最重要和次重要），并基于信道条件采用不同的差错保护方式。

在下一节中我们给出一个无线视频流媒体系统跨层优化的例子。该方法综合考虑了应用层和物理层的有效抽象和使用基于应用的目标函数，从而对多层进行联合优化。进一步，我们不但通过测试环境中的实验给出了跨层优化的性能增益，而且我们还探索了跨层优化性能增益和额外的计算和通信成本之间的折中。

6.5　无线视频流媒体跨层优化架构

让我们考虑一个应用场景，一个基站向其覆盖范围内的 K 个移动用户传送流媒体视频[12]。在 CLA 中，应用层、数据链路层和物理层被联合优化（见图 6-7）。跨层优化被同时应用于所有的用户来分配资源，并利用了多用户的多样性[18]。

简而言之，跨层优化周期按照以下方式工作（后面小节中将详细介绍）：首先，CLO 获取各层参数的抽象。物理层和数据链路层的抽象基于两种状态的 Gilbert-Elliott 模型的转移概率。应用层的抽象基于速率失真分布，该分布包括帧的大小和接收端每种类型帧丢失的期望失真[20]。在抽象过程后，CLO 通过选择视频源速率的最优值（应用层）、分配时隙（数据链路层）和调制方案（物理层）来优化系统。

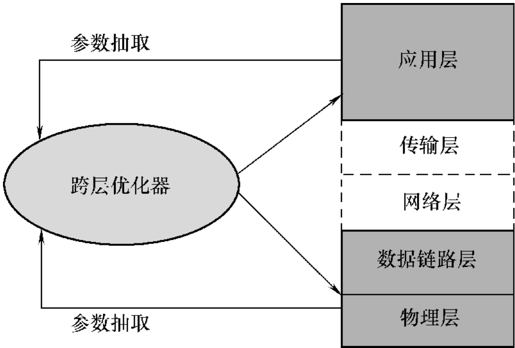


图6-7　视频流媒体跨层架构

峰值信噪比（Peak SNR，PSNR）是一个可以表示用户感知视频质量的可量化参数，该参数可以作为视频流媒体系统优化的标准。CLO根据测量接收端期望的PSNR可以找到用户感知的视频质量的最大值。目标函数可以用不同的方式定义，比如，根据特定用户的PSNR（也许是蜂窝内视频质量最差的用户）或者根据蜂窝内所有用户的平均PSNR。一旦CLO选好了最佳的参数值，它就将它们分发到所有的独立层，这些独立层负责把这些参数翻译回真正的操作模式。

6.5.1　抽象层参数

CLO使用应用层、数据链路层和物理层的抽象。

应用层被抽象成速率失真分布[20]，该分布描述了信道质量变化对用户感知的视频质量的影响。速率失真分布由以下几个元素构成：

1）速率矢量，由GOP中每个帧的大小构成。

2）失真参数，描述了接收端（以平均平方差错表示）对每个帧丢失的失真，其中假设接收端显示最近解码的帧而不是丢失的帧（见图6-8）。失真参数在编码时计算，存储在流媒体服务器中，并且随着视频流一起发送给CLO。

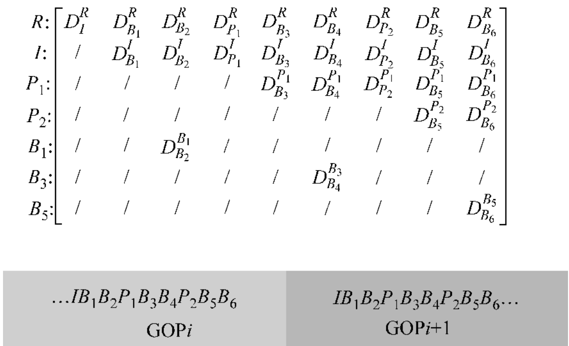


图6-8　GOP的率失真矩阵

每一行显示的是最近解码的帧而不是丢失帧；R 表示前一个 GOP 中最后解码的帧；矩阵元素表示失真 D，以及在各自位置上（以下标表示）的帧丢失和显示最近解码的帧（以上标表示）。

物理层使用 Gilbert-Ellilot（GE）模型进行抽象，该模型将无线信道的包差错变化分成两个状态：G（好状态）和 B（差状态）。在好状态下，假定包被正确而及时地接收，而在差状态下，假定包丢失。用 p 表示从 G 状态到 B 状态的转移概率，q 表示从 B 状态到 G 状态的转移概率。描述每个用户信道的转移概率（p 和 q）根据传输数据速率、传输包差错率、数据包的大小和信道连贯时间 4 个参数来计算，参见 Peng 等人的论文[12]。

物理层一个更抽象的表示可以由 GE 模型按照下面的方式得到。假设一个 GOP 由 15 帧组成，每帧只传输一次。帧丢失可以分为 16 种不同的模式。模式 1 表示 I 帧中至少一个包丢失，因此 I 帧在接收端无法解码。由于帧的依赖关系，当前 GOP 中的所有帧都无法解码，它们将被前一个 GOP 中最后解码的帧代替。模式 2 表示 I 帧中所有的包都被正确接收，但是 P_1 帧中至少一个包丢失。其他情况可以类推。模式 16 表示没有包丢失。给定 GE 模型中的转移概率（p 和 q），每帧丢失模式 p_i 的概率可以由式（6-1）推导出

$$
\begin{gathered}
p_1 = 1 - P_G(1-p)^{(n_1-1)};\\
p_2 = P_G(1-p)^{(n_1-1)} - P_G(1-p)^{(n_1+n_2-1)};\\
\cdots\\
p_i = P_G(1-p)^{(n_1+\cdots+n_{i-1}-1)} - P_G(1-p)^{(n_1+\cdots+n_i-1)};\\
\cdots\\
p_{15} = P_G(1-p)^{(n_1+\cdots+n_{14}-1)} - P_G(1-p)^{(n_1+\cdots+n_{15}-1)};\\
p_{16} = P_G(1-p)^{(n_1+\cdots+n_{15}-1)}
\end{gathered}
\tag{6-1}
$$

其中 P_G 表示在好状态下的稳态概率，而 n_i（$i=1$，$\cdots15$）表示由速率矢量和包大小[12]决定的第 i 帧中包的数量。

式（6-1）中的帧丢失模式概率在假定每帧只传输一次的前提下可以推导得出。然而，当分配给一个用户的传输速率要大于视频源速率时，最重要的帧可以被传输多次来减少帧丢失率。在重复传输的情况下，当一个包的至少一份备份被正确接收时，这个包就被认为成功接收了。当传输速率不足以用来重传所有包时，那些最重要的包被重传直到有足够的传输速率。

6.5.2　优化

在每个 GOP 开始，CLO 选择应用层、数据链路层和物理层的参数值，以获得最好的用户感知的视频质量。这需要 CLO 根据 PSNR 为每个用户、每个选择的参数评估接收端的期望的视频质量。PSNR 可以通过以下两种方式之一获得：

1）使用失真信息在接收端计算期望的重构质量。假定每个丢失模式下，帧丢失模式概率为 p_i，得到的重构失真为 D_i，则期望的重构失真 D_{exp} 可以表示为：

$$D_{exp} = \sum_{i=1}^{16} p_i D_i \text{。}$$

2）当没有失真信息时计算期望可解码帧数。这个期望 PSNR 的近似值准确性稍差，得到的是次优的配置。

然后，一旦得到了每个用户的视频质量，CLO 就可以通过最大化可以定义的目标函数的方法优化网络层参数。比如，根据所有用户中最差的视频质量和所有用户的平均视频质量。

6.5.3　性能和成本分析

让我们考虑这样一个场景，3 个用户观看一些从位于基站的流媒体服务器传输的典型的测试视频，如母亲和女儿（MD）、Carphone（CP）和 Foreman（FM）。所有的视频都是 QCIF 格式（176x144），帧率 30f/s。采用 MPEG-4 编码成两种不同的源速率，100kbit/s 和 200kbit/s。每个 GOP 有 15 帧，包括 1 个 I 帧和 14 个 P 帧。系统的传输能力假定为 300k symbol/s。采用两种不同的调制方案，BPSK 和 QPSK，整个速率分别为 300kbit/s 和 600kbit/s。每个用户的可能速率为｛0，100，150，200，300｝。3 个用户之间有 72 种可能的速率分配。

图 6-9 比较了下面 3 种情况的性能：

1）使用跨层优化，CLO 使用来自速率失真参数（RD）（有 RD 的 CLO）的期望 PSNR 来评估视频质量。

2）使用跨层优化，CLO 使用 ENDEF（无 RD 的 CLO）来评估视频质量。

3）不使用跨层优化（无 CLO）。

在 3 种情况下，对应不同的 SNR 范围计算平均 PSNR 的累计分布函数（Cumulative Distributed Function，CDF）。在第一种情况下，所有用户的信道条件都不好（SNR 范围从 0 ~ 5dB）。采用有 RD 的 CLO 时的平均 PSNR 比无 RD 的 CLO 时的要高 2dB 左右。在第二种情况下，仿真中的 SNR 随机地从 0 ~ 25dB 之间选取。在第三种情况下，所有用户有相同的好的信道条件（SNR 在 20 ~25dB 之间）。在所有 3 种情况下我们观察到，当使用有 RD 端信息的跨层优化时的平均 PSNR 要比无优化时的要提高 2dB 左右，而使用无 RD 端信息的 CLO 时的性能介于两者之间。然而，在第三种情况下，使用无 RD 的 CLO 的性能非常接近于无优化时的性能，这是因为可解码帧和得到的 PSNR 之间的相关度较低[21]。

我们的分析表明，使用期望的可解码帧数当和无 CLO 比较时仍然可以提供

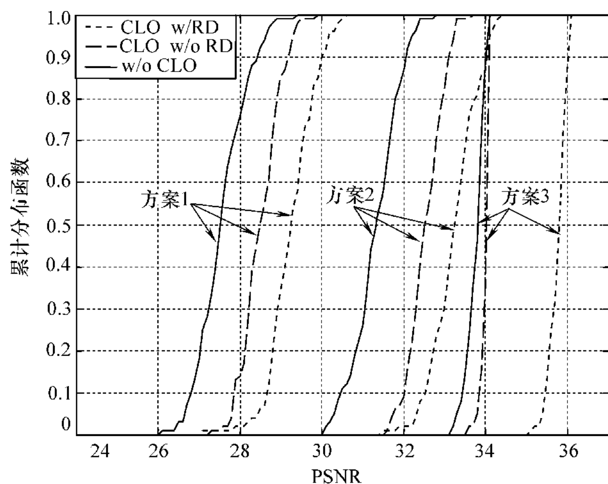


图 6-9　平均 PSNR 的累计分布函数

有效的增益，特别是在 SNR 较低的信道情况下。使用 RD 分布的优化能够提供更大的性能增益，因为计算的期望视频质量更为准确。然而，应该考虑服务器端由于传输 RD 分布导致的额外通信成本。图 6-10 给出了不同源速率下的额外通信开销，假设 GOP 最开始是 I 帧，后面全是 P 帧。额外开销相对较低，但是随着 GOP 中的帧数增加会线性增加。

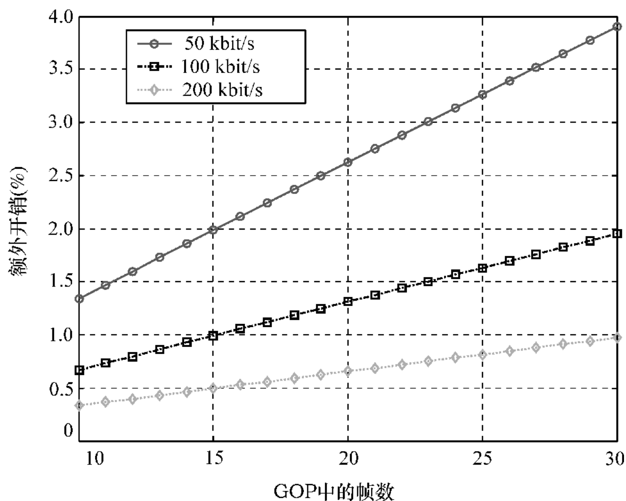


图 6-10　RD 分布的额外流量开销

6.6 小结

跨层设计（CLD）是一种新的方法，将来很有可能用在提高通信网络的设计和管理中。然而，目前仍然存在一些需要解决的技术问题。

到目前为止，CLD 已经被成功地应用到一些场景中，并且表明通过跨层优化协议栈可以获得性能增益。然而，以前的研究经常忽略了跨层优化引入的额外成本，这些成本在资源受限的系统中尤其需要考虑。评估成本和性能之间的这些折中需要开发一些系统级的方法论和分析工具。

另外一个基本的问题是跨层优化的实现。集中式的优化器可以最好地找到最优设置。然而在实时系统中这种方式几乎不可能实现，因此出现了分布式的设计，每层中的本地优化器以比全局或集中式优化器更高的频率做出决策。

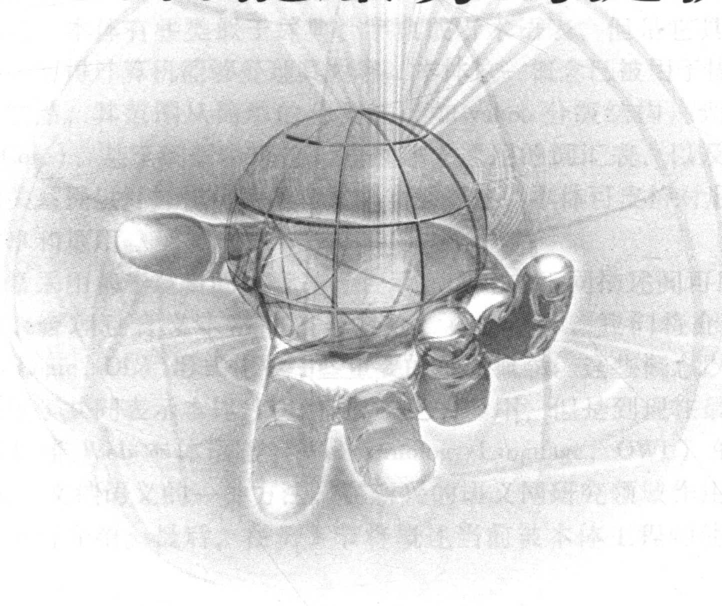
为了允许实时地计算各层的参数，必须正确地抽象描述各层设置的参数。在优化质量和优化时间上存在折中。优化器输入的参数越多，优化的结构可能就越好，但是计算的代价高。一个基本的研究问题就是如何找到能够获得近优设置的最少的抽象参数。

选择合适的优化参数是另一个基本问题，尤其是在多媒体应用中。在这些应用中，用户感知的质量是主观的，因此难以测量。此外，当多个应用同时运行，并且共享网络资源时，为了保证所有用户的满意度必须定义一个常用的目标函数。Khan 等人在文献［22］中提出，语音通信中使用的平均评价分也许是一个合适的指标，但是需要映射成其他应用类。

应用跨层优化通常能够提高网络能力和增加服务的用户数。然而当网络在一个分布式和高度动态变化的环境中运行时，很难完全保证在所有情况下满足 QoS 要求。临时性的缺少资源将可能要求降低某些用户的服务质量，甚至中断服务。当资源无法满足所有限制时，必须保证用户间的公平。定义公平的资源分配策略是另一个热门的研究领域。

第2部分

基础智能服务的提供



第 7 章　本　　体

Marko Luther

近年来，大家开始关注如何理解与海量信息组织相关的科学原理。逐渐流行的一个方法就是寻求通过本体的构建和执行实现信息的组织。本体就是对概念化的一种形式化描述[1]。在本体中，概念依据描述某领域知识的公理和定义进行分类[2]。因此，本体有些类似于辞典、字典或者术语表，但是它具有更多细节描述和结构，使得计算机能够处理其内容。本体这一概念已被用于描述具有多级结构的人工制品，其范围从简单的分类法（如 yahoo 分级结构）到元数据模式（如 Dublin Core），甚至到逻辑理论。通过建立通用的词汇表，以及对相关术语及术语间的关系提供计算机可用的含义丰富的描述，本体可支持对形式化表示后的知识的共享和重用。

本体通常采用基于逻辑的语言进行表示。因此在不同描述间可区分出详细、准确、一致、合理且含义丰富的差异之处。在 7.1 节，我们将介绍描述逻辑（Description Logic，DL）形式化中一些重要的基础概念。这些概念以及其相关的推理服务被用于共同表示本体。DL 有很广泛的应用，但是到现在最知名的应用是作为支撑标准 Web 本体语言（Web Ontology Language，OWL）的形式方法。OWL 作为表示文档语义的一种方法，在新兴的语义网研究领域作用显著。OWL 将在 7.2 节进行介绍。最后，在 7.3 节将概述当前被本体工程师使用的方法和工具。

7.1　描述逻辑

描述逻辑（DL）[3]是一组基于一阶描述逻辑子集的通用知识表示形式，其使用的一阶描述逻辑子集的推理功能止于对逻辑推论的验证。描述逻辑的发展可解决在实现知识表示的特殊方式（比如说在 20 世纪 70 年代开发的语义网络和框架系统）中发现的语义问题。

在 DL 领域的研究早期被表示为术语系统，其含义指表示语言主要用于建立某领域的基础术语。后来，研究的重点开始转移至形成概念的连接符，名称也改为概念逻辑。现在，研究框架逐渐建立在术语描述逻辑之上，突出了底层逻辑系统的重要性。

基于理论与实践的紧密结合，DL的研究已成功覆盖了推理的形式化和计算特性（见7.1.3节）以及基于DL系统的实现（见7.1.4节）。这些系统已在多个应用领域证实了其有用之处（见7.1.5节），并被广泛地接受。特别是，作为表现力丰富的语言的应用已成为语义网本体语言（见7.2节）的形式化基础，并在近几年被更广泛的接受。

7.1.1 基础描述语言

对于DL而言，最基础的内容在于概念间的包含关系，这些关系可定义属性的继承性以及其导出的概念分级。DL的典型特征在于其可以表示在概念间成立的其他种类的关系。然而，概念间的关系越复杂，越难于准确地刻画暗指的推论，以及完整有效地计算暗示的关系。

DL方式清晰区分了内涵知识和外延知识，其中内涵知识表示在某领域中通常一成不变的通用知识，而外延知识则涉及特定问题。内涵知识通常使用描述通用概念的术语公理进行阐述，这些公理构成所谓的T-box；而外延知识使用关于个体的一组命题公理进行阐述，这些公理的集合即所谓的A-box。

使用一元谓词符号标识的原子概念有时也被称之为基础概念，它和使用二元谓词符号标识的原子角色一起构成定义T-box的基本组成。复杂术语通过使用更小集合的构造符（由概念和角色组成），可由基本符号构成。在左侧包含原子概念的等式可为复杂描述引入符号化名称，因此也称之为定义。

$$\text{人} \equiv \text{男人} \sqcup \text{女人}$$

$$\text{妈妈} \equiv \text{女人} \sqcap \exists \text{子女}.\ \text{人}$$

上述定义可定义一个概念的必要和充分的条件，而陈述蕴含关系的原始定义则可用于描述只包括必要条件的部分。

$$\text{女人} \sqsubseteq \text{人}$$

用于复杂描述C的公理$C \sqsubseteq D$常被称之为通用概念蕴含公理（GCI）。

描述逻辑可使用其提供的由角色和概念组成的操作加以区分。定语语言（$\mathcal{AL}$）[4]引入了原子概念（A）、通用概念（$\top$）、底层概念（$\bot$）和否定符，其中否定符只应用于原子概念（$\neg A$）、相交（$C \sqcap D$）、通用数值限制（$\forall R.C$）以及有限的存在量化关系（$\exists R.\top$）。另一个有名的DL是简单命题描述逻辑（$\mathcal{ALC}$），它对应于通过将语法限制于包含两个变量的公式所得的一阶逻辑片段[3]。$\mathcal{ALC}$对$\mathcal{AL}$进行了扩展，补充了完全补集（如概念否定）、逻辑或（$C \sqcup D$）和完全的存在量化关系（$\exists R.C$）。结构语言（$\mathcal{FL}^-$）也叫做结构化描述逻辑，通过在$\mathcal{AL}$中禁用原子否定演化而来。而作为从实用出发的最小集语言，$\mathcal{FL}_0$则通过在$\mathcal{FL}^-$禁用有限的存在量化关系而来。

总的来说，DL语言通过构建记忆名称加以分类，其中在记忆名称中对逻辑

的精确表达进行了编码。在系统名称中加入了对应字母可表示为扩展附加构造符的 DL 语言。为了避免在表示意义丰富的 DL 中出现过长的名称，引入了缩略语 *S* 来表示 $\mathscr{ALCR}^+$。也就是说，这种 DL 通过形象地增加闭合原语[5] 来扩展 $\mathscr{ALC}$，其中闭合原语可通过增加后缀 $\mathscr{R}^+$ 表示（见表 7-1）。

表 7-1　描述逻辑 *S*

构造符名称	语　法	实　例
原子概念	A	人
通用概念	⊤	⊤
包含	(C⊑D)	男人⊑人
相等	(C≡D)	子女≡儿童
原子角色	R	子女
转换角色	*Trans* (R)	朋友
联合	C⊓D	人⊓女人
析取（U）	C⊔D	男人⊔女人
否定（C）	¬C	¬男人
有关存在的条件（E）	∃R. C	∃子女. 人
数量约束	∀R. C	∀子女. 女人

A-box 包含由单个公理（有时也叫做论据）组成的外延语言。例如

alice ∈ 女人

表示 alice 是个女人。同理，

alice 子女 bob

特指 bob 是 alice 的一个孩子。第一类断言称之为概念断言，而第二类断言则叫做角色断言。基于A-box 和 T-box 的推理服务将在下一章节加以介绍。这些服务将用于计算术语公理和断言公理的分类。这些公理在本节中进行了介绍，并在图 7-1 中加以描述。

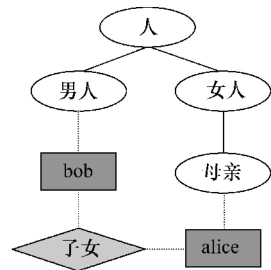


图 7-1　分类法

7.1.2　推理服务

基于本体的推论关系到如何基于形式化概念、角色和个体描述归纳结论。关于概念和个体的隐式知识可通过合理完整的推理算法自动推理获取。这些算法在大量 DL（见 7.1.3 节）中加以应用。特别是，概念间的关系以及个体与概念间的实例关系发挥了重要作用。

1. 标准推理

在 DL 中，对概念表示的基本推理是包含，它表示一个概念的标识符是另一

个概念的子集。另一个典型的在设计期重要的推理任务是决定某一描述是否具备可满足性（比如不冲突）。一个分类的过程可通过包含关系，将术语的概念组织成类似格子的结构。A-box推理则关系到判断其断言集是否具有一致性（如是否有一个模型）和一个特定个体是否是给定概念描述的一个实例。在表7-2中列举了本段落描述的基础推理任务以及其他对知识库操作的典型查询。然而，为了实际需要，许多DL系统支持附加的查询操作（比如对个体或概念名称集合的检索，见7.1.4节）。

在开发阶段，关于逻辑一致性的查询对于保证结果知识库的质量特别重要。纯粹的建模会在T-box中产生不一致的概念定义（如描述个体的空集合），这种不一致的定义会导致影响每个判定的不一致数据库，使得本体推论毫无用处。同样，当所给个体限制与T-box中相反时，不一致性也会出现在一个知识库的A-box中。

需要注意的是，一致性检查和分类过程都会关系到整个T-box，导致对知识库的显著影响。即使知识库被认为是一致的（比如通过只加入合理概念来完成知识采集过程），重新分类也会造成T-box的明显重构。

表7-2 标准推理服务

推理任务	描述	影响
概念一致性	对象集是否用空概念表述	T-box
概念包含	由两个概念表述的一组对象间是否存在子集关系	
一致性检查	查找T-box中提及的所有不一致概念，不一致概念可能是建模不当或数据收集错误的结果	
分类	决定T-box相关概念的父类和子类，概念的父类是在包含概念的T-box中最具体的概念名称，概念子类则是在包含概念的T-box中最概括的概念名称	
A-box一致性	对应T-box的A-box中提及的现实条件是否过强，如它们是否相互抵触	A-box
实例测试	某个个体所代表的对象是否是一组由某个特定查询概念描述的对象的成员，如果是，则这个个体被称为一个查询概念的实例	
实例检索	从A-box中找出所有符合条件的个体，这些个体代表的对象，可被证明为一组由某个特定查询概念描述的对象的成员	
个体直接型	从T-box中找出最具体的概念名称，且给定个体是该T-box的一个实例	

2. 开放和闭合世界的假设

DL的一个突出特点是它们提出了开放世界假设（Open World Assumption，OWA）。一个数据库实例只能表示一个解释，在这个解释中模式的类和关系通过实例中的对象和三元组加以说明。而一个A-box则表示多种不同的解释，即其所有的模型。因此，在一个数据库实例中的信息缺乏可解释为否定信息，而在A-box中的信息缺乏只意味着某种知识的缺失。随之而来的就是，数据库的信息

通常被认为是完整的，而 A-box 中的信息一般被视为不完整。A-box 具有开放世界语义的观点在 DL 技术中得到了清晰的反映，DL 也使用 OWA 进行推理。这意味着“无法证明是正确的东西并非一定是错误的”。因此，一个否定回答可代表“对于给定信息无法进行验证”。

如图 7-2 所示，关于 miller 不在，同时缺乏另一个人的信息时并不能解释为“那里没有人”。即便所有与角色“together”相关的可知个体都是概念“同事”的实例，基于 OWA 原则也无法得到如下结论，即个体“miller”符合概念“工作中”的说明。OWA 保证了有关附加断言（如“miller 与 wallace $\sqsubseteq \neg$ 同事”）的继承限制关系的单一性，这也许在以后会加入 A-box。

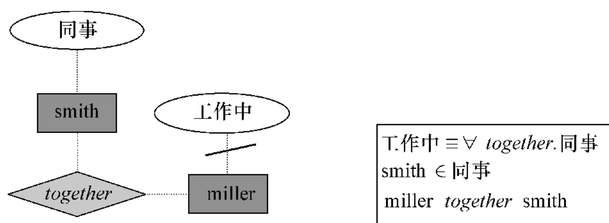


图 7-2　开放世界假设

相对而言，闭合世界假设（Closed World Assumption，CWA）则规定所有不可知或无法证明正确的事情都假设为错误的。CWA 源于 20 世纪 70 年代后期的 AI 和数据库的研究，并且同样的基础假设在当今的编程语言（如 Prolog）和大多数数据库设计中一样成立。此假设的有用之处在于其允许在信息缺乏的情况下进行附加推理。然而，有时当一些未知因素判断有误时，一个假设可能也是错误的。

7.1.3　语言扩展

将概念和角色的结构分离成简单的由术语构成的操作符开启了对一个语言组的广泛分析。结果则是，DL 族可能是最被深入理解的一组知识表示形式。表示语言的表达能力和推理复杂性间的权衡使其计算空间已得到深入的分析。

在上一章节，我们介绍了基础的 DL 概念和角色构造符。然而现在使用的大多数 DL 都提供了更强的语言特征，这点经常通过为系统名称增加附加字母进行表示（见表 7-3）。字母 $\mathscr{H}$ 用于表示具有多个父类情况下的角色等级关系。如果某种语言支持逆角色，则用字母 $\mathscr{I}$ 表示；并且如果允许功能限制特性，则需增加字母 $\mathscr{F}$。字母 $\mathscr{N}$ 代表简单的数量限定，而 $\mathscr{Q}$ 用于有条件的数量限定。字母 $\mathscr{O}$ 则表示针对某个名词概念外延规格的语言构造符。

表7-3　对 S 的扩展

结　构　名	语　　法	举　　例	系　　统
角色等级	$R \subseteq S$	父亲⊆人	H
逆角色	R^-	监督 = 监督人	I
功能角色	$Funct\ (R)$	父亲	F
非定性数量限定	$\geqslant nR$ $\leqslant nR$	≥∃儿童	N
定性数量限定	$\geqslant nR.\ C$ $\leqslant nR.\ C$	≥∃儿童．女性	Q
名词概念	$I_1 \cup \cdots \cup I_n$	{红，绿，蓝}	O
具体领域集	$u_1,\ \cdots,\ u_n.\ P$	temp. （>20）	(D)

为了克服大多数 DL 对某些特定领域（如类似 numbers 和 strings 等的具体数据类型、空间区域或定性的时间间隔等）的有限支持，Baader 和 Hanschke 定义了将具体领域集成于 DL 的通用方法，方法中通过对系统名称增加（$\mathcal{D}$）进行表示[6]。

如图7-3所示[9]，$\mathcal{S}$家族中主要成员包括$\mathcal{SH}$[7](在$\mathcal{ALCR}^+$基础上扩展了角色等级)、$\mathcal{SHIF}$[8](在$\mathcal{ALCR}^+$基础上扩展了数量限定、逆向角色和(≤1 R)形式的数量限定）和$\mathcal{SHIQ}$（也被称之为$\mathcal{ALC}$团，在$\mathcal{ALCR}^+$基础上扩展了角色等级、逆向角色和有条件的数量限定)。

由于对扩展逆向角色的$\mathcal{S}$的推理复杂性仍属于最坏情况的多项式空间（PSpace)[5]，其上限并非健全的 w. r. t 扩展（见表7-3)。而推理过程对于$\mathcal{SHOQ}$（$\mathcal{D}$）[10]、$\mathcal{SHOI}$[11]甚至是没有名词的系统（如$\mathcal{SHIF}$和$\mathcal{SHIQ}$[9]）而言，其复杂性也属于最坏情况的确定性指数时间（ExpTime)。对于包含名词和具体领域的系统（如$\mathcal{SHOIN}$($\mathcal{D}$)[12]，即对$\mathcal{SHIQ}$加以无条件的数量限定，并进行名词和具体领域扩展)，其推理复杂度更高，即是最坏情况的非确定性指数时间（NExpTime)。

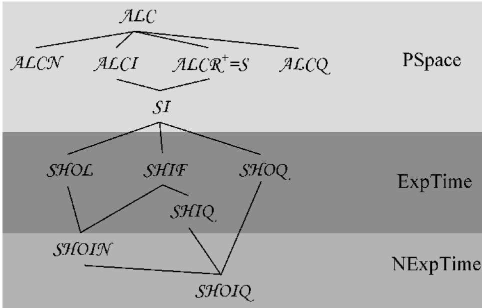


图7-3　重要的描述逻辑

尽管增加一个特性并不会改变某系统的最坏情况复杂度，但其可能影响实际的实现。比如说，在能力非常强的 DL 中的逆向特性就使得几个重要的优化技术有效性大大降低[13]。在对$\mathcal{SHOQ}$或$\mathcal{SHIQ}$的优化算法进行扩展，使之成为$\mathcal{SHOIQ}$（如对$\mathcal{SHOIN}$进行有条件的数量限定扩展）过程中，由于名词、数量

限定和逆向角色交互[14]引起的难点直到最近才得到解决[15]。

7.1.4　描述逻辑系统

所有 DL 系统都源自 KL-ONE[16]，其推动了从语义网络到合理 DL 的转变。与表现力更强的后继者（如 LOOM[17]）一样，KL-ONE 中的包含关系被定为是结构化的。然而，虽然这点是合理的，但是用于逻辑语义的结构化算法却是不完整的。后来，人们开始重点关注对完整推理服务的需求，从而导致了 KRIS[18]系统的出现。在该系统中引入了基于缓存和繁殖的优化分类算法。在更近一段时间，对于经典的表微积分的专门化直接导致完整的推理算法实际应用于表现力丰富但难于处理的 DL 中[4,7]。然而事实证明，推理的难于处理（类似于最坏情况下的非多项式复杂度）并不会阻碍 DL 在实际中的使用价值。在实际应用中，它提供了精致的优化技术，比如惰式演变、吸收和依赖性控制的回溯等[7]。当前具有强大表现力的完整有效的 DL 系统包括 FaCT[19]、RACER[20]和 Pellet[21]。这些系统多多少少可提供在 7.1.2 节中提及的所有推理服务，下面做一下更详细的介绍。

FaCT（Fast Classification of Terminologies，术语的快速分类）是针对 $\mathcal{SHIQ}$ 逻辑[19]中表达的术语，使用通用 Lisp 编写的一个高度优化的 DL 分类器。它支持使用包含 CGI 的知识库的推理，但不能处理个体或具体的数据类型领域。此外，FaCT 还不支持多个 T-box，也没有提供删除概念定义的机制。目前已出现采用 C++重新实现的 FaCT，即 FaCT++[22]。它实现了 $\mathcal{SHIF}(\mathcal{D}-)$ 逻辑，并且建立在引入新的优化机制的内部架构之上。虽然 FaCT++实现了 A-box 推理，但对具体领域的支持仅限于整型和字符串型（因此称之为 $\mathcal{D}-$）。

RACER 是另一个用通用 Lisp 实现的高度优化的表微积分推理器。它支持加入到 FaCT 中的所有优化技术，以及处理数目限制和 A-box 推理的附加优化措施。这些附加优化措施是基于对给定知识库和查询进行静态分析来实现动态选择的。类似于 FaCT，RACER 也实现了 $\mathcal{SHIQ}(\mathcal{D}-)$，同时也支持 A-box 推理服务和 A-box 断言回缩，以及多个 T-box 和 A-box。此外，RACER 还为代数推理提供了便利条件，包括用于处理整型限制、实数线性多项式方程、复数非线性多变量多项式方程和字符串等式、不等式等具体领域。然而，RACER 并不支持针对名词的完整推理。在 RACER 中将计数类中的个体转化为不相交的原子概念，仅仅类似于名词。

Pellet 是一个针对 $\mathcal{SHIN}(\mathcal{D})$ 和 $\mathcal{SHON}(\mathcal{D})$ 的可靠且完整的表推理器，也是针对 $\mathcal{SHION}(\mathcal{D})$ 的一个可靠但不完整的推理器[21]。此外，Pellet 通过使用为 $\mathcal{SHOQ}(\mathcal{D})$ 开发的算法，为名词提供了可靠且完整的推理器[12]。它是用 Java 实现的，并且引入了不少特性。Pellet 可以为内置的原始 XML 模式数

据类型、联合 A-box 查询和继承限定检查提供本体分析、修复以及数据类型推理。

7.1.5 应用

DL 用于大量的应用和具体系统中[3,23]。DL 最早的应用领域之一就是软件工程，当时在 AT&T 中加以使用，以实现用于详细说明大型软件系统事实的软件信息系统。其他在该领域的成功应用还包括在复杂系统开发中通过验证组件配置的特定属性为开发者提供帮助。

医疗是从 20 世纪 80 年代初期就开始开发专家系统的领域。DL 中的有关分类学的表示和推理能力激发了医疗知识超大型本体的构建和维护。除了诸如自然语言处理和数据库管理等深入的应用领域，还有不少基于标记语言应用的重要成就被用于从语义层面注解 Web 结构的信息内容[24]。DL 在语义网应用程序设计中的应用将在下一章进行阐述。

7.2 Web 本体语言

曾有人预言，本体将在语义网研究中充当关键的角色。W3C 也已经勾勒出了第二代 Web 的愿景，即 Web 资源更易于被自动化过程访问[25]。语义 Web 的关键环节就是使用机器可以理解的描述资源内容的元数据对 Web 资源进行标注，而使用本体提供共享的、精确定义的条目供元数据使用。上述要求需要对 Web 标记语言进行扩展以便于内容的描述和基于 Web 本体的发展，如 XML 模式、资源描述框架（Resource Description Framework，RDF）和 RDF 模式（ RDFS）。

RDF 是一种用于标记 Web 资源的通用语言，而 RDFS 则是一种对属性及资源种类进行定义的模式语言。其中 RDFS 特别被公认为是一种本体表示语言，它允许采用有类型的层次结构来组织词汇表，而这种方式是通过定义类及属性（二进制基本关系）、属性的取值范围和领域约束、类的实例以及子类与子属性关系来实现的。然而，RDFS 的表达能力仍相当有限，对推理支持仅限于约束检查。它缺乏必要的构造符，以定义类的不相交性、类的布尔值合并以及性能的特性（比如传递性），并且缺乏对属性可取或必取的确定值数目进行限制（比如基础约束）的可能性。

为了解决 RDFS 有限表达能力的问题，W3C 基于早先提出的 DAML + OIL[27]定义了更富表达能力的 Web 本体语言（OWL[26]），其中 DAML + OIL 本身就是由基于框架的美国提议 DAML + ONT 和基于 DL 的欧洲语言 OIL 融合而成（见图 7-4）。为了形成一个标准的、可被广泛接受的语义网本体语言，需要提供定义明确的语法以及具有足够表达能力和推理能力的语义。形式化语义及推

理能力通常是通过将本体语言映射到一个已有的逻辑形式上来实现的。以 OWL 为例，选用具有较强表达能力的描述逻辑（已在上一章节进行介绍）作为语义基础，使得基于已有 DL 推理器的应用可实现有效推理。

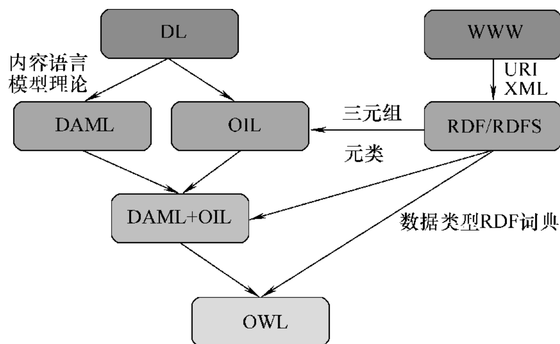


图 7-4　OWL 演变

在层次结构上，OWL 位于 XML、RDF 以及 RDFS 之上[28]。语法上，OWL 可以被看做是 RDFS 的一个特殊方言（如 OWL 的个体都是用 RDF 描述来定义的）。合法的 OWL 文档必然也是合法的 RDFS 文档（反过来则不成立）。然而，OWL 使用的“XML/RDF”语法可读性较差，这一点在下一章节的例子中有对应描述。因此，人们定义了更具可读性的 OWL 抽象语法[29]。

理想状态下，OWL 语义应由 RDFS 扩展而来，即 OWL 使用 RDF 关于类及属性的定义，但需补充语言原语以支持更丰富的表达能力。然而，如果这些从 DL 发展而来的附加原语与 RDFS 的更高阶进行融合将引入不可控的计算属性。因此，OWL 被定义为 3 类不同的子语言（见 7.2.1 节），每个子语言完成不相容需求的不同方面。OWL Lite 和 OWL DL 是具有 RDF 语法的富含表达能力的描述逻辑。而 OWL Full 则通过覆盖所有的 RDFS 具备了最强大的表达能力和语法自由度，但存在严重的逻辑问题。在 OWL Full 的 RDFS 部分中缺乏对类和个体的语义区分导致了核心推理问题的不可判定性（比如可表现 Russell 的矛盾观点）。

7.2.1　语言元素

OWL 本体基本上就是 RDF 文档，其中 OWL 的头部是个 rdf：RDF 元素，用于指定多个命名空间。本体自身开始于 owl：Ontology 元素（包括注释、版本控制以及通过 owl：import 语句对其他本体的引用）下的一个断言集合。输入语句是可传递的文本形式的蕴含语义。下面的片段是引用了本体 gender. owl 的本体 humans. owl 的合法头部：

```
<? xml version =`1.0′? >
<rdf:RDF
    xmlns:rdfs =`http://www.w3.org/2000/01/rdf-schema#′
    xmlns:owl =`http://www.w3.org/2002/07/owl#′
    xmlns:agent =`http://localhost/Ontologies/0.1/agent.owl#′
```

```
      xmlns:rdf ='http://www.w3.org/1999/02/22-rdf-syntax-ns#'
      xmlns:xsd ='http://www.w3.org/2001/XMLSchema#'
      xmlns ='http://localhost/humans.owl#'
   xml:base ='http://localhost/humans.owl'>
   <owl:Ontology rdf:about =''>
      <owl:imports rdf:resource ='http://localhost/gender.owl'/>
   </owl:Ontology>
```

类由 owl：Class 元素进行定义，其中 owl：Class 为 rdfs：Class 的子类。下面的片段定义 Woman 为 Human 的子类，但与类 Man 不相交（实现中使用了类构造器 owl：subclassOf 和 owl：disjointWith）。

```
<owl:Class rdf = about ='#Woman'>
   <rdfs:SubClassOf rdf:resource ='#Human'/>
   <owl:disjointWith rdf:resource ='#Man'/>
</owl:Class>
```

类的等价关系通过使用 owl：equivalentClass 元素进行表达。进一步说，使用类的组成构造符 owl：complementOf、owl：disjointWith 以及 owl：intersectionOf 可以定义类的布尔合并。最终需要有两个预定义的类 owl：Thing 和 owl：Nothing 分别对应描述逻辑的顶层及底层元素（见 7.1.1 节）。

属性可通过使用 owl：Restriction 和 owl：onProperty 等构造符对类定义加以限制。例如，下面的 OWL 片段将 Mother 概念定义为有孩子的女人（也就是类 Woman 与 has_ child 属性施加的限制条件的交集，该限制条件限制所有的成员需与概念 Human 的个体相关）。

```
<owl:Class rdf:ID ='Mother'>
    <owl:equivalentClass>
        <owl:Class>
            <owl:intersectionOf rdf:parseType ='Collection'>
                <owl:Class rdf:ID ='Woman'/>
                <owl:Restriction>
                    <owl:someValuesFrom rdf:resource ='#Human'/>
                    <owl:onProperty>
                        <owl:ObjectProperty rdf:ID ='child'/>
                    </owl: onProperty >
                </owl:Restriction>
            </owl:intersectionOf>
        </owl:Class>
    </owl:equivalentClass>
```

```
</owl:Class>
```

OWL 的类限制声明 owl：someValuesFrom 对应现存数量，而 owl：allValues-From 则对应全局数量。其他的限制声明包括基数限制 owl：minCardinality、owl：maxCardinality 和 owl：cardinality。在 OWL 中存在两种属性，其中对象属性将一个对象作为子类绑定到另一个对象，而数据类型属性则将一个对象关联到一种数据类型。属性可通过 owl：ObjectProperty 和 owl：DatatypeProperty 两个构造符进行定义，可能还包括取值范围限制（基于 rdfs：range 和 rdfs：domain 实现）。此外，通过使用 OWL 元素 inverseOf、subPropertyOf、TransitiveProperty、SymmetricProperty、FunctionalProperty 以及 equivalentProperty，一个属性可与其逆向关系关联。一个子属性可被描述为对称的、功能性的或可传递的，此外，还可定义属性的等价性。下面的片段即对象属性 child 将定义为 relative 的子属性，这也同时将 humans 与逆向属性 parent 相关联。

```
<owl:ObjectProperty rdf:about='#child'>
        <rdfs:range rdf:resource='#Human'/>
        <rdfs:domain rdf:resource='#Human'/>
        <owl:inverseOf rdf:resource='#parent'/>
        <rdfs:subPropertyOf rdf:resource='#relative'/>
</owl:ObjectProperty >
```

枚举，也称为名词或值集合，是使用构造符 owl：one Of 进行定义的。如下面例子所示，枚举可通过罗列所有元素对类进行详细说明。

```
<owl:Class rdf:ID=' Gender'>
    <owl:one of rdf:parseType=' Collection'>
       <owl:Thing rdf:about='#male'/>
       <owl:Thing rdf:about='#female'/>
    </owl:oneOf>
</owl:Class>
```

类的个体，有时也叫实例，跟 RDF 中的声明相同。例如，下面的片段就说明了个体 alice 是一个女人，并且有一个儿子叫 bob。

```
<Woman rdf:ID=' alice'>
      <has_child>
         <Man rdf:ID=' bob'/>
      </has_child>
</Woman>
```

在 OWL 语言参考文档[26]中包含了这里介绍的语言构造符的更多细节，并且描述了用于说明数据类型、标记、个体不相关性和版本信息的其他构造符。需要

注意的是，OWL 并没有进行惟一名字假设。相反，它认为两个不同的对象名字在 DL 语义下代表不同的东西。

7.2.2 子语言

按照表达能力从低到高，OWL 可分为 3 种子语言：OWL Lite、OWL DL 和 OWL Full[30]（见图 7-5）。其中 OWL Full 的表达能力最强，它允许使用在上一节中提及的所有构造符，并覆盖所有的 RDFS。由于在子语言 RDFS 中没有明确区分类和个体，OWL Full 引入了一些高阶特性，从而导致了核心推理问题的不可判定性[9]。而 OWL DL 作为一阶逻辑的可判定片段，使用了与 OWL Full 相同的词汇集，但增加了一些附加语法限制。有了这些限制，OWL DL 可对应描述逻辑 $\mathcal{SHOIN}(\mathcal{D})$，在 ExpTime 情况下可判定（见 7.1.3 节）。OWL Lite 则是 OWL DL 的一部分，旨在为初级使用者提供一个更易表达的有效的语言特性子集。OWL Lite 的表达能力大致等同于具有 ExpTime 最坏情况复杂度的 DL 语言 $\mathcal{SHIF}(\mathcal{D})$。与 OWL DL 不同，OWL Lite 摒弃了在描述或类公理中有关合并、取补和个体的构造符。另外，它将基数限制为 0 或 1，且将嵌套描述限制为概念标识。然而，这些限制也使得表达能力相对削弱。借助于间接的和语法上的技巧，除了包含个体或大于 1 的基数的描述，所有的 OWL DL 都可在 OWL Lite 中进行表达[28]。

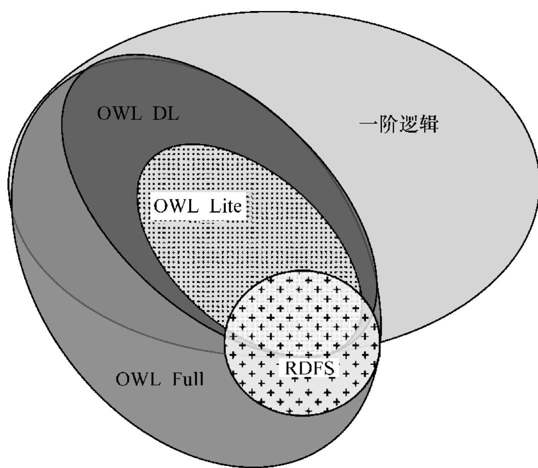


图 7-5 OWL 子语言

对于相对传统的 DL 研究而言，名词是非标准化的。所以直到最近 Horrocks 和 Sattler 研究成果[15]的出现，针对完整 OWL DL 语言的实用且完整的算法在很长一段时间都是未知的。然而这些成果仍然没有解决如何实现一个可靠、完整且高效的推理系统以支持 OWL 的 XML 模式数据类型，而这些都在传统 DL 语言的标准具体领域（整数、实数和字符串）之外。其中逻辑的包含关系如图7-6所示。

7.2.3 基于规则的扩展

由于 OWL DL 关注于可判定性，因此在其中仍有很多事情不能加以表示。

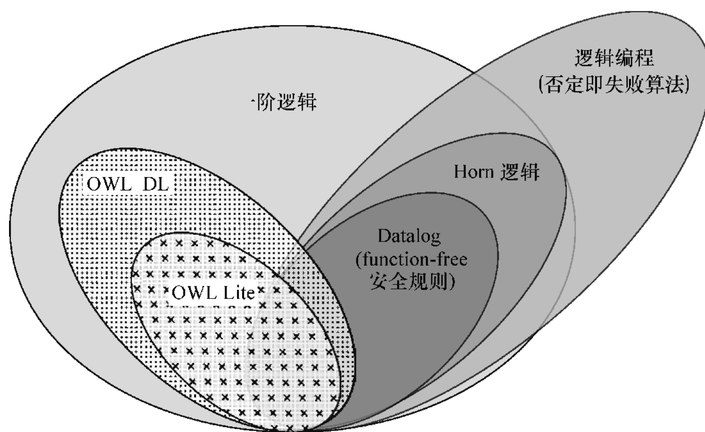


图 7-6　逻辑容器

其中很多涉及到属性链，也就是表示多属性间约束的能力（例如“叔叔”属性是由“父亲”及“兄弟”组合而成）。在一个本体的 T-box 和 A-box 中，除了基于 DL 的推理外，与较高级别的本体交互以克服 OWL 表达能力的限制也是必要的。因此，针对本体的查询和规则语言应运而生，这样“叔叔”属性可以借用变量来匹配 A-box 的实例，具体表示如下：

Parent(？x,？y)和 brother(？y, ？z)⇒uncle(？x, ？z)

然而，需要重点强调的是，即便具备 DL 的表达能力，查询回答也不能像在数据库框架下那样简单地简化为模型校验。实际上，在 DL 设置下的查询回答一样需要像逻辑推导那样的推理手段[31]。当前基于 DL 的系统通常只提供能力较弱的查询语言，这些语言只限于检索（查找给定概念的实例）、实现（判断某个体是哪个最特殊概念的实例）和实例化（采用布尔查询的形式询问一对个体是否为一个给定角色的实例），特别是系统并不支持使用变量公式化的查询。最新的研究成果表明合取的查询语言可以针对 DL 系统丰富关于在查询中使用变量的特定限定，从而为传统 DL 系统的一个缺陷提供解决方案。完全去除这些限定会引出问题，特别是在变量被用于在查询中强制循环的情况下更严重[31]。现在已提出多个使用查询语言增强本体推理的建议。DQL[32]就是一种在 DAML + OIL 本体中支持查询-回答的形式化语言，而 OWL-QL[33]则是 DQL 为适应 OWL 所做的修改版。推理器 RACER 也在最近扩展了其新的 RACER 查询语言 nRQL[34]，该语言可支持在任意概念和角色表达式中所绑定变量的提取。

用于在 RDF/S 描述库中查询和说明观点的说明性语言也已在一些文献中出现，它们包括 RQL[35]、RDQL[36]和 TRIPLE[37]（详尽比较见 Haase 等人的文章[38]）。然而，同这些语言不同，基于本体的查询语言可支持查询-回答对话，在对话中可以使用自动推理方法推导答案。

规则系统包括规则库（如一个无序的 if-then 规则集）、运行中的内存（如一组以文字形式表示的事实）和实现前转和后转链接的规则引擎。规则中的“if”部分是一系列模式，用于说明引发规则可用的事实。将事实与模式匹配的过程叫模式匹配。以知识表示格式使用的规则系统具有简易性和模块化的优点。由于规则被认为是相互独立的（至少在一定程度上），因此规则库易于维护和升级。此外，具有精确优先的冲突解决策略的通用规则推理实现了简单的默认推理。然而，默认推理本质上是非单调和不可判定的（见 Reiter、McCarthy 和 McDermott 在非单调推理中的工作成果）。此外，大多数规则系统缺乏清晰定义的语义。因此，通过表现特性（如规则库的一致性或推导过程的可终止和可汇合）来确保这些系统的可靠性是个复杂课题[39]。一个（静态）分析必须覆盖整个推理系统，因为实现的因果关系同时基于规则引擎和具体的规则库。特别是规则冲突（比如两个具有不一致结论的应用规则）难以避免且十分棘手。这些情况使得规则集的含义对冲突解决策略的选择十分敏感。这正如 Patrik Winston 所说“遗留控制的优点变成了失去控制的缺点”[40]。

通过使用专门化的规则集实现 OWL 推理是个计算量很大的工作，并且关于概念的推理必须间接地通过创建典型实例来实现[41]。此外，虽然听上去很完整，但这种基于规则的方式并不完整（甚至对 OWL Lite[42]而言）。Jena 指南中指出这种方式只适用于包含基于轻量级有规则本体的实例推理的应用[41]。

很明显，DL 与规则推理的融合是大势所趋，因为包含复杂 DL 表达式的规则的通用形式比 Horn 规则明显具有更强的表达能力[17]。特别是，这些通用规则能够表示关于存在和否定的信息，而这些在（可判定）Horn 逻辑的一阶谓词逻辑子类中是无法表达的（见图 7-7）。然而，类似规则可用于模拟角色值映射这样的紧组合也很容易造成一些有趣推理问题的不可判定[43]。一个典型的例子就是 Motik 等[44]提及的可能导致不可终止。Grosof 等人[45]分析了描述逻辑与单调规则系统间可能的冲突。基于这项工作以及早期在复合推理方面的工作[17,46]，看上去一个可行的做法是在 DL 的表达能力和功能无关的 Horn 规则中取交集。然而，在语义网规则语言 SWRL[47]的建议中提到的将 DL 与 Horn 规则直接合并，可用于模拟可明确导致不可判定的图灵机[46]。这种在表现力上的限制在描述逻辑程序[43]中已公式化，以禁止连续出现关于存在的可定量知识。判定性也可以通过将规则限制成所谓 DL-safe 的情况加以保持，这种情况下需要每个规则中的变量在规则体中都出现于非 DL 原子中[44]。

此外，DL 推理也可以与规则推理实现松耦合，即首先推导 DL 结果，然后再将规则应用于结果之上。然而在这种情况下，一些结论即便在语义遗传情况下也难以推导出来。比如，这种方法就无法从 A-box 断言“abel 是个 GoodChild 或 BadChild”、规则“如果是 GoodChild（x）就是 Child（x）”和“如果是 BadChild

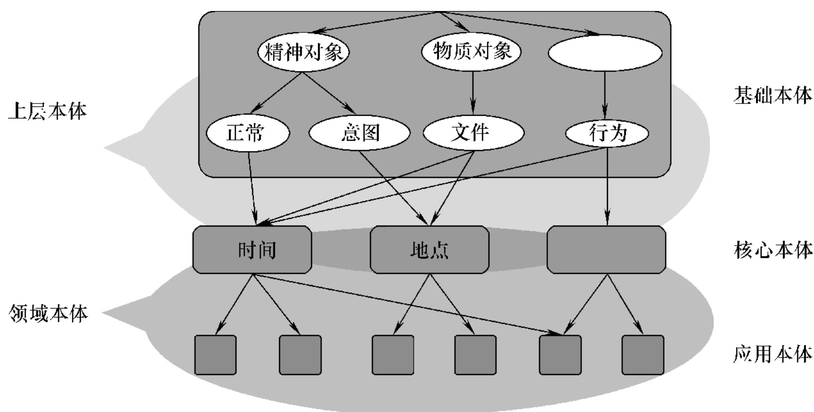


图 7-7　本体分层

（x）就是 Child（x）”中推导出事实“Child（abel）”。

对于规则可行的扩展包括前面所提的 OWL 规则语言 SWRL 和 Racer 内置的查询语言 nRQL，SWRL 允许事件-条件-动作形式的规则公式在基于推导结论的应用中可以自动触发行为，但通常这是不可判定的。相反，nRQL 无法直接触发行为，但却是可判定的，且紧密集成于核心 DL 推理引擎中。然而，nRQL 提供了一种发布-订购的机制，这使得知识库发生变化时可及时通知客户。

7.2.4　语言缺陷

OWL 中 owl：import 构造符的功能在模块化方面的表现是不尽如人意的，它将所有关联的本体融合到一个独立的逻辑空间，很自然地导致了一个完全扁平的本体[48]。另外，本体的输入在 OWL 上也不是安全的。例如，将一个 OWL DL 文档输入到另一个文档可能会生成一个 OWL Full 文档。一个可行的解决方案就是扩展 OWL，以实现在关联的本体中进行分布式静态推理[49]。

另一个问题与 OWL 的开放世界假设（OWA）[3,40]相关。在 W3C 的 OWL 需求建议[50]中，表达闭合世界的能力被看作是一个目标，而不是一个需求。OWA 的优势在于推理的单调性。如果附加信息可用，则先前的推理结果依然成立。然而其缺点是一些结论无法从守恒的 OWA 中导出。一种可行的解决方式是将闭合世界特征集成到在本地闭合世界假设条件（Local Closed World Assumption，LCW）[51]下的代理社区中出现的开放世界系统中。借助 LCW，闭合世界的信息可以在已知完备信息的子集中获得，并且仍然允许将其他信息看作是未知的。然而，如何使 LCW 与 OWL 的 DL 方式兼容仍有待研究。

当前可用的推理支持也存在一些缺陷，大部分可用的推理器并不完全支持 OWL。例如，流行的 RACER 系统就缺少对 owl：import 语句的全部支持。这就意

味着一些相关联的本体并不能被直接加载。此外，RACER 没有实现增量推理[52]，这使得上下文变化后，需要进行一次完整的重推理。同样地，被大部分 DL 推理工具和编辑器实现的标准描述逻辑实现组中的 DIG 接口[53]也不支持一个本体特定部分的删除。这种现象在动态应用中出现得相对频繁，这也使得在修改属性后需要重新加载（或重分类）所有的本体。

最后，增加复杂的类定义或对多个实例进行保存和查询，在许多应用领域都是很常见的，但这会降低已有推理器的性能，甚至到不可用的程度。现在已有一些研究在尝试解决这一问题，比如利用 hyprid DL-DB 架构的 Instance Store[11] 和 LAS[54]。然而，这些解决方案仍存在限制。例如，Instance Store 就只能处理不包含任何角色断言的数据。

7.3 本体工程

为一个相当大的领域构建本体很容易成为一件繁杂的工作。这项宏大的工作如果没有合适的方法论是不会成功的。这一方法论会像其他软件工程产品一样指导本体的开发。本体工程指的是一系列的行为，涉及的内容包括本体的开发流程、本体的生存期、构建本体的方法和方法论以及所使用的工具和支持的语言。

7.3.1 设计原则

下面是 Gruber[55] 所制定的设计原则，它解决了已被认为是与任何本体的开发流程都密切相关的问题：

1）范围：本体开发应该集中在选定的领域。

2）明确性：术语所要表达的含义应能有效传达。因此，本体应易于应用开发者使用，并且应支持实际的、有意义的、直观的和简单的询问。一个具备完整表达力的详细本体，如果相比大多数应用所需要的必要的详细程度还要复杂的话，用处就不大了。

3）一致性：给定的定义应该具有一致性。它们不仅在逻辑上保持一致，而且本体非形式化的部分也应该与形式化部分保持一致。

4）可扩展性：本体的设计理念应具有可扩展性。本体中应便于增加新的名词而无需修改原有的定义。这样，一些特殊领域的本体就能够构建在已经定义好的、更加通用的本体之上。

5）最小的编码偏差：表现形式的选择不应以便于标记和实现为目的。本体应该尽可能独立于本体所使用的应用和描述本体的语言。

6）最小的本体约定：为了使本体尽可能可重复使用，本体应在尽可能支持特定用途的同时，减小对现实世界的假设。

1. 语言的选择

用于描述本体的语言限制了本体本身的表达能力以及其对应用的适用性和可重用性。例如，像OWL Full这样的语言通常是不可判定且难以处理的。表达的一致性不是自然而然达到的，还需要人们处理其他一些未完成的重要推导工作。如果此问题非常明显，就应该参考已有的标准本体，使用一些标准的语言。IST的Esperonto项目组最近发表了一份关于如何选择本体描述语言的综述报告[56]。

2. 命名习惯

每一项本体的开发都必须严格遵从一个命名的约定。这一点不仅使得本体更易于阅读和理解，并且有助于避免一些通常的建模错误[57]。下面列出了一些公认的命名约定：

1）使用英语来表达名字。

2）本体应该用简短的、描述性的单数词语表示，需要使用小写字母，并且采用“owl”作为文件扩展名（比如“agent. owl”）。

3）概念应该用首字母大写的单数名词表示，即使是一个概念代表一组事务（比如“Wine”）。

4）组成实体名字的名字之间以下横线作为分隔符连接，而不是使用RDF中采用的Intercap方式或XML元内容框架[58]。因此可以使用“Business _ meeting”，而不能采用“BusinessMeeting”。

5）实体和属性采用单数名词（而不是动词），或者单数名词序列，使用小写字母（比如“chianti”和“colleague”）。

6）属性名字不能添加“has”或者“is_ ”作为前缀，不能添加“of”作为后缀，也不能采用动词形式。因此可以使用“parent”，而不能使用“parenting”或“parents”。

7.3.2　结构化

不同领域的应用都对应各自的本体，因而产生了针对不同范畴的本体。针对这一情况，IST的WonderWeb项目[59]给出了层次化的本体架构。下层本体对上层本体提出表示需求，而上层本体为下层本体提供设计原则。

我们可将本体分为三大类别（见图7-7）。在最高层定义了所谓的基础本体，包含高层的、独立于领域的概念，比如对象、时间和过程（如广泛覆盖）。这些概念基于源自语言学、哲学和数学的形式化法则。第二层是核心本体，在这一层构建了可重用性非常好的信息建模原语的工具箱。核心本体提供了特定领域的基础设施（如中度覆盖），位于基础本体和特定应用本体之间，特殊化了基础本体并有助于集成特定应用的知识。这样看来，一个核心本体为一系列的特定应用本体提供了基础设施。特定应用本体位于最底层，并将某个感兴趣的领域的概念和

属性关联起来（如它们具备小范围覆盖）。

基础本体和核心本体的抽象部分有时也被称作上层本体。通过提供标准的知识表示原语库，上层本体促进了分布式信息系统的语义互操作性。此外，与上层本体保持一致可以为应用本体提供可靠的支撑，并有助于避免由于解释不准确造成的术语和概念上的歧义。

我们提倡根据属性来源和领域的约束条件将属性结构化。这种办法在归并语义关联的属性和在增加新的属性时对约束条件进行自动验证方面是行之有效的。

7.3.3 开发流程

在本体开发中，并不存在惟一最优的方法论或流程。这也意味着总会存在不同的办法。然而在本节中，我们将讨论一种实用的开发流程，它可以在一定的开发和维护代价的前提下使本体在各应用场景下取得最优可用性。

一个有效的本体开发流程必然是迭代的，在每次迭代过程中对本体不断优化。在迭代过程中，最重要的是采用应用场景对本体进行评估，以确保它的概念反映了那些场景的实际情况，同时又能够广泛支持不同的场景。

Akkermans 等[60]和 Noy 等[61]根据通用的知识工程方式提出了一个开发流程。如图7-8所示，这个流程包含4个主要的阶段：

1）开始阶段：第一步决定了本体针对的领域和范畴以及本体设计需求。如果可行，这一步将考虑重用本领域已有的本体。通过使用应用场景，将建立一个包含重要概念及其属性和相互关系的枚举列表。

2）改进阶段：这一步将使用类似 OWL DL 的形式化描述语言对开始阶段得到的结果以及本体评估和维护中的所得进行形式化。类和类的分层架构将使用该语言进行设计。这个设计过程既可以从最通用的概念开始（自顶向下），也可以从最特定的概念开始（自底向上）。领域专家将在这个阶段工作中发挥主导作用。

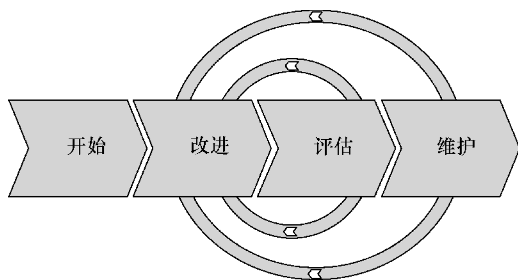


图7-8　本体开发流程

3）评估阶段：评估阶段的作用是验证已开发本体的有用性。显然，应使用开始阶段定义的需求对本体进行检查，但更重要的是本体的用途应该在应用场景上下文中进行检验。这可能是一个演示过程。对概念的最佳验证是将本体提供给应用开发者，并观察他们的使用方式。从这些开发者收集上来的反馈和经验对于进一步改进本体而言是非常有价值的输入。

4）维护阶段：最后一个阶段主要是一个组织阶段，但是也同样非常重要，因为本体很有可能会在实际使用中不断发展。因此，设计开发合适的规则和工具来更新本体（删除、插入、修改概念或者关联），并且指派负责的团队和人员进行维护是非常重要的。针对更新本体的请求，必须制定清晰的流程，比如通过网站来发布本体的新版本。

7.3.4　标准本体

集成大规模的、抽象的、针对特定领域场景的本体是一个巨大的挑战，并且通常是一件低效和费时的任务。然而，轻量级的上层本体能够提供对一个通用词汇表的访问，这个通用词汇表使得不同知识表示间的互操作作成为可能。

1. 核心本体

为了对数量众多的现有核心本体进行分类，首先有必要明确统一的标准（比如它们的类型、定量的大小和语种）。下面的本体被定义为 OWL DL 上层本体：语言学和认知工程描述性本体 DOLCE（Lite）[62]、OpenCyc 空间本体[63]、REI 策略本体[64]、FOAF（朋友的朋友）[65]、普适应用标准本体 SOUPA[66]、由 DAML-Time 推导而来的 OWL-Time[67] 和 IEEE 建议的上层混合本体 SUMO[68]。此外，还有一些核心本体要么与移动服务没有直接关系（如 COBRA-ONT[69]），要么尚未公开（如 CONON[70]）。此外，服务核心本体 OWL-S[71] 将在下一章详述。

接下来我们将总结对上述可得且相关的核心本体的评价。其他一些对上层本体的评测主要针对美国政府和军方[72]或空间领域[73]。

DOLCE 是一阶本体，仅包含上层本体，因此其适用范围广泛（至少其精简版本 DOLCE Lite 如此）。DOLCE Lite 是十大最复杂的 OWL DL 本体之一。OpenCyc 空间本体和 REI 策略本体都能够划分为比 DOLCE 更多的领域特定的核心本体，分别集中在空间信息、安全接入和控制方面。尽管其大小可观（大约包括 5000 个概念和角色），OpenCyc 本体架构的部分内容经常被用作其他项目的基础。而 REI 策略本体则仅由一个基本的分类学架构组成。因此，它的全部内容都可以被集成到其他本体中。FOAF 是关于代理的一个基础描述，尽管只提供了原语概念，其 OWL DL版本（FOAF Lite）也被应用于不同的项目中。与 FOAF 一样，OWL-Time 本体拥有很好的架构，它的精简版也经常被引用到领域特定的上下文中。而 SOUPA包含了一些其他的本体（比如 FOAF、OWL-Lime 和 OpenCyc 空间本体），重点吸收了它们的部分词表，但是并没有将其直接引入。SUMO 本体包含了最广泛的、最抽象的概念以及对原语概念的全面分类。因为 SUMO 具有完整的一阶基础，其 OWL 解释属于不可判定部分的 OWL Full。此外，SUMO 中的几处不一致在近几年使用自动一阶法则证明器 Vampire[74] 的过程中也已被发现。

2. MobiLife 上下文本体

IST 的 MobiLife 项目[75]定位于改进个人和群组日常生活中所使用的移动应用和服务。项目的研究内容之一是设计一个通用框架，支持根据指定上下文为相关用户提供服务并相应地调整自身功能。因此，上下文几乎被当作交互时所能得到的全部信息片。上下文的处理是由一个管理框架完成的，该框架提供了呈现、维护、共享、保护、推理和查询上下文信息的有效方法[76]。在 MobiLife 项目中，已经决定依靠本体技术来表达和推断高层次的、定性的上下文信息。但是，由于本体的缺陷在于处理大规模数据，以下描述的上下文本体并没有作为上下文各方面主要的表示格式。

MobiLife 核心本体提供了代理、空间和时间实体的概念描述，以及对多种设备和个人时间表的分类描述。此外，情景组件本体定义了特定应用的概念，以描述移动用户的典型场景。MobiLife 本体定义了超过 800 个概念、属性和个体，并将其分割为 9 个模块，写入了 OWL DL 的可判定片段（如 OWL DL）。各个组件本体之间的相互关系可参见图 7-9。

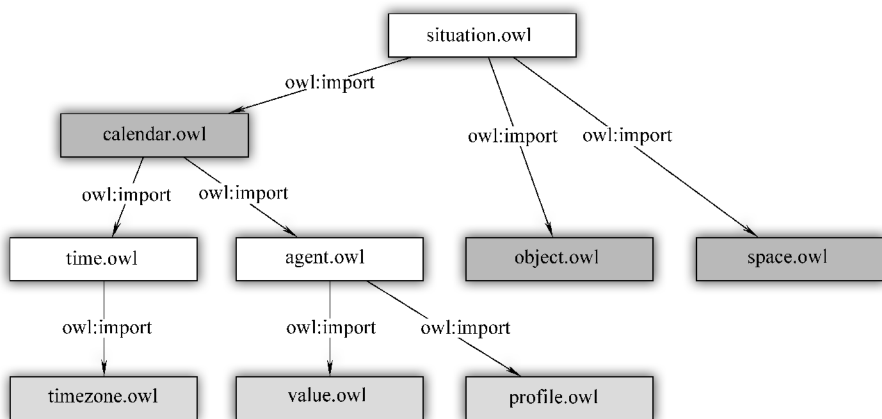


图 7-9 MobiLife 组件本体之间的相互关系

Agent. owl 模块涵盖了描述个人、组织和群组的概念。它学习了 vCard 标准[77]和 FOAF 词表[78]，提供了描述个人联系信息的词汇表，并定义了它们之间的关系[79]。time. owl 本体建立在 OWL-S（time-entry. owl[80]）中使用的标准 time. owl 本体的子集上，并集成了如 Allen 代数[81]中定义的时间间隔之间的所有定性关系。类似地，space. owl 组件本体提供了通用的位置词表，集成了区域连接演算 RCC[82]。其他模块则提供了个人信息相关的词汇表（profile. owl）、定性值（values. owl）和时区相关的词汇表（timezone. owl）。最后，本特定领域的情景本体（situation. owl）通过公理化定义的概念，对 MobiLife 场景特定的可推导

情景上下文进行了分类。图 7-10 即为情景本体的一个片段。

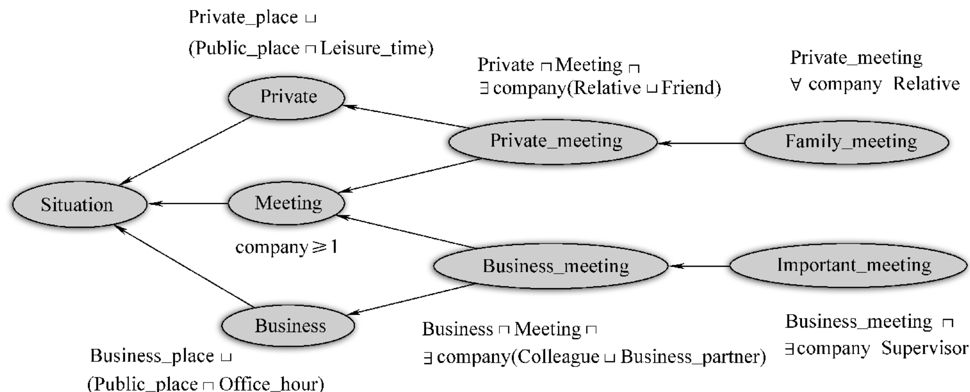


图 7-10　情景本体

7.3.5　开发环境

由于构建本体是一个费时的工作，因此在大多数本体工程任务中都使用软件工具帮助开发者进行开发。当前的本体开发环境不仅在构建本体的过程中，可为知识工程师提供简单的编辑环境，而且提供图形化的本体导航、验证、调试、版本管理以及连接外部资源至本体的解决方案。在下面的章节中，我们将介绍 3 种最具创新性的 OWL 本体开发环境，即 Protégé[61]、SWOOP[49] 和 OntoTrack[83]。

1. Protégé

Protégé 系统是基于知识的系统开发环境，其发展已超过 10 年[61]。Protégé 起初是为医学领域设计的一项小应用，但是却发展成为具有通用意义的知识工程工具集。它建立在 HP 的 Jena 程序库[41] 基础上，并于近期加入了对 OWL 的增强支持[84]。当前版本 Protégé 3.1 可以运行在多种平台之上，支持增强的图形用户界面，可以与标准存储格式（如关系数据库、XML 和 RDF）进行交互，并已经被数量众多的个人和研究机构使用。Protégé 可以与几个推理引擎集成（比如 RACER[20]；另见 7.1.4 节），以便为知识工程师提供支持。此系统由美国 Stanford SMI 免费提供、开发和维护。

2. SWOOP

语义网本体概览与精读 SWOOP[49] 是基于 Java 开发的图形化本体编辑器，在其设计和使用中采用了浏览器隐喻。它提供了基于超级链接的本体实体导航、往返移动用的历史按钮以及可存储并用于后续参考的书签。除了 OWL 的 RDF/XML 语法，它还支持 OWL DL[29] 的正式抽象语法。与其他应用环境一样，SWOOP 同样来自于多种本体的支持，它通过与 Pellet 推理器[21] 的紧密耦合实现

了对验证的支持和类别校验（见 7.1.4 节）。SWOOP 的高级功能包括强大的版本管理支持和运行健全完整的联合 A-box 查询（使用 RDQL 语言[36]描述），其中版本管理支持功能中变化集可链接到相应的由 Annotea 插件[49]生成的协作注释消息，通过将变化集转变成独立的 $\mathscr{E}$ 连接（$\mathscr{E}$-connected）本体[86]实现本体的自动区分[85]。最近该环境中增加了一个新的调试模式，用于支持白箱（glass-box）和黑箱（black-box）调试，将 Pellet DL 推理器的内部工作流以可读易懂的形式呈现出来，以帮助使用者理解检测出不一致性的原因[87,88]。

3. OntoTrack

其他的本体编辑工具通常使用两个功能独立的接口分别用于编辑和浏览，这种方式会影响查找和浏览的速度。OntoTrack[83]采用统一的界面克服了这一缺点。该系统将一个复杂的图形布局，与鼠标编辑功能相结合，同时增加了解释组件[89]，其中鼠标编辑功能可针对有效浏览和操作大规模本体进行优化。OntoTrack引擎提供了复杂的布局算法，支持继承类的自动展开和收拢、放大/缩小、缩略图和平移功能。其最具创新性的特点是瞬间反馈功能。通过与外部推理器同步每个编辑步骤，可检测并显示相关的建模结果。现在，OntoTrack 能够处理大部分 OWL Lite，除了个体、数据类型属性和注释属性。

7.4 小结

本章所介绍的本体技术是知识表示、形式化逻辑和自动推理领域超过 30 年研究的成果。现在，基于该技术开发的最吸引人的应用是在新兴的语义网研究中的资源范围。然而在未来该技术可能会在上下文感知的移动服务方面取得更大成功。特别是在移动服务提供过程中集成可判定的推理机制将尤为重要。通过这种方法，所提供服务规格的一致性将可以得到验证，并且通过使用形式化的通用知识，可推导出隐式的上下文信息。这样，移动通信系统就能够检测并排除错误的服务描述并提供可靠的情景分析。这两方面对于实现主动服务提供场景都是需解决的重要问题。

然而，将基于本体的表示与推理支持进行集成仍是一件复杂的事情，这需要多个领域长期的研究经验。首先，即便本体的开发领域在最近取得一些长足的进展，该领域仍然缺少足够和可靠的工具支持。然而实际上，即便是诸如 Protégé 这样广泛使用的本体开发工具，目前也无法提供全语言支持[61]。

此外，OWL 在维持核心推理任务的可判定性的同时，却仅能提供受限的表达能力，这经常仅仅能够提供抽象场景描述。设计上下文感知的未来移动服务和应用就意味着需要接受挑战，更好地定义具体场景。然而，能否仅使用表达能力受限的 OWL 来解决问题，目前仍没有答案。

第8章　语义服务

Massimo Paolucci

8.1　挑战与机遇

近几年，为用户随时随地提供互联网接入已成为移动计算发展的挑战和动力。一些相关方面的研究和努力已大大增强了移动用户间的互通性，并最终推动了类似i-mode等新型服务的出现。i-mode 可使移动用户通过其移动电话无限制地接入 Web。通过使用 i-mode[1]，用户可以访问 Web 页面，读取相关信息，并针对自己的生活做出相应的决定。

目前，随着随时随地的互通性得到了保障，移动计算的发展方向已转为如何使移动平台处理相关信息，以更好地在日益复杂的生活中服务于用户（本章中，“移动平台”一词是个通用词汇，以代表移动电话、PDA 及其他任何可以为用户提供计算能力的手持设备）。对于移动电话而言，其功能不仅仅是检索网页，供用户阅读，它们还应能够收集相应信息，并在用户需要时显示出来。现在的问题已经从向移动平台发送信息，转变为如何使移动平台主动参与到用户的生活中，比如说为改善用户行为提供关键性的信息。

上述未来的远景正逐步变成现实。这些变化所需的一些服务已经存在，比如说，现在已经有相关服务可以提醒我们该去赶飞机了，或者飞机已被取消，又或者告诉我们计划搭乘列车的晚点信息。目前，针对移动用户提供相关服务才刚刚开始，相信在不久的将来伴随技术进步和服务的普遍应用，这一服务会快速展开。

然而在上述远景成为现实前，还有很多重要的技术挑战需要得到解决。第一，移动平台需要能够“发现”用户所需的服务。一旦识别这些服务，移动平台需要启动它们：提供所需的输入参数，解释输出结果，并将结果提交给用户。通常情况下，基本没有哪个单独的服务可以满足用户的需要，更有可能的情况是需要将现有多个服务组合成复杂服务，为解决共同的问题服务。

新近推出的 Web 服务标准（比如 WSDL[2]、SOAP[3] 和 UDDI[4]）为解决移动平台的服务问题提供了一个自然的起始点。Web 服务技术旨在定义一个“面向服务的体系架构[5]”。在此架构中，信息的交换不依赖于网页，而是通过服务进行。移动计算的技术难题在于如何将相应技术应用于计算能力有限的平台中，

以及在通信有限、通信可靠性差并不断剧烈变化的上下文环境中如何完成相关操作等问题。同驻留在连接情况良好、资源充足的计算机集群的Web服务及应用相比较，移动平台一般是由用户随身携带的，他们位置灵活，可随处移动。当一个用户从位于地下的车站走到地面上的站点时，他的智能电话必须适应一个全新的计算环境：由于在地下站点可用的服务在地面上无法访问，智能电话必须能够将其替换为在地面站点可访问的服务。

此外，如何实现更高层次的自动化，使用户摆脱管理多服务间互操作的负担，也对移动计算环境下泛在服务的动态性提出了一定的要求。虽然在B2B的上下文环境中指派程序员实现一个与服务交互的客户端软件是个简单易行（尽管昂贵）的方法，但这种方式在移动计算环境下是不可取的。即便是savvier用户，也不会在赶车或出差时花费时间编写访问服务的客户端。泛在计算成功的核心关键点在于提供智能客户端的能力。这种客户端可以针对变化的环境实现自动配置，并可以挖掘所需的信息。

遗憾的是，由于Web服务标准无法表示服务与客户端间交互所隐藏的语义，它们难以满足支持高层次自动化的要求。本章的主旨在于说明挖掘语义信息是移动计算未来发展的关键点。此外，我们提出Web服务技术应建立在本体（见第7章）和语义网[6]技术的基础之上。这些技术可提供相应语言，利用逻辑公理和完备的逻辑推理机制来描述对应的信息以及不同类信息的关系。而公理和相关推理机制可以共同支持对和相应信息一致的结论的推导。最终，作为语义网理论基础的逻辑推理系统，使得客户端可以决定如何使用一个服务以及如何解释来自服务的消息。

本章中我们将描述语义网服务技术（即如何将Web服务技术应用于语义网）的发展现状，并详细讨论如何基于该技术实现对泛在服务的访问。本章的成果集中于泛在服务在移动计算环境中的应用前景，以支持泛在服务所需的动态交互层面。虽然该前景建立在现有技术的基础之上，仍有很多内容有待实现。对应章节的组织如下：8.2节提供了一个真实性场景，讨论与现有服务交互的问题；8.3节和8.4节则分别讨论Web服务技术以及引入语义的必要性；8.5节则描述如何利用OWL-S[7]语言融合Web服务和语义，以解决上文重点提到的发现和调用问题；8.6节将简要回顾其他语义网服务技术；最后，8.7节将讨论开放问题，并在8.8节给出我们的结论。

8.2 实际体验

我们提出了很多针对通过移动平台可访问的泛在服务的愿景。这些想法并不是关于科学会议的未来性想法，相反它们正在快速成为现实。目前，通过

SMS 或 i-mode 已经可以访问多种服务，比如列车票付费和航班座位登记。为了更好地理解相应技术，我们将在本节分析一个具体实例，例中一个游客将在意大利和德国使用手机接收列车时刻信息。该实例的描述建立在 Ferrovie dello Stato 公司（以下简称 FS）和 Deutscher Bahn 公司（以下简称 DB）提供的现有服务基础之上。其中 FS 公司是意大利一家铁路公司，而 DB 公司则是德国一家铁路公司。

对于用户而言，首要的问题是发现相应的服务。这些信息虽然可以通过铁路公司印刷的小册子或公司网站获取，但却没有一个“一站式”的解决方案，使得用户可以查询所需类型的服务。换句话说，针对移动平台的现有服务，还缺少一个相关的目录（有很多类似 fastfind. com 的网站可以用于查找某公司提供的某些 SMS 服务，但必须是要事先知道这些公司名。对于“查找列车时刻表”这样的问题，是无法找到相关服务的）。

一旦找到了相应的服务，游客的下一个问题就是如何调用该服务，特别是游客需要编辑相关信息，并发送至服务端。毫无疑问，两个不同的铁路时刻表服务要求的消息格式是有所区别的。比如意大利的服务要求的消息格式如下：“Da Roma a Milano 27/06/2008 10∶30”（来自 http：//www. trenitalia. it/it/area_ clienti/sms/index. html），该消息表示获取在 2008 年 6 月 27 日 10：30 后从罗马直达米兰的列车时刻信息。而德国的服务则需要采用如下格式的消息：“BAHNMIX Berlin Hamburg 27. 06. 08 10∶30”（来自 http：//www. bahnmix. de/mix_ mobil. php），上述消息用于获取在相同时间段从柏林到汉堡的列车时刻信息。重要的是这两条消息无法混合在一起，因为德国的服务将无法回复意大利的服务所采用的类似“Da…a…”格式的消息，而意大利的服务也理解不了德国的服务所采用的类似“BAHNMIX…”的消息。接下来的第三个问题是如何在相应位置提供正确的参数。比如“BAHNMIX Hamburg Berlin 27. 06. 08 10：30”消息会提供与用户要求相反方向的列车时刻信息。最终，用户还需要解释请求的结果，并获取相应的列车信息。

让事情更糟糕的是，许多服务可能要求一条以上的消息。比如，某座位预订服务需要至少 3 条消息：第一条用于找到相应车次；第二条用于判断是否满员；而第三条则用于支付预订的费用。这些消息中的每一条都需要认真编辑，并且与前面的消息紧密相关。这种情况使得用户必须记住可能被设备记录的所有信息。

很明显，上述服务有很突出的问题。用户应该只需要关注旅程，如目的地、出发和到达时间等，而其他所有细节应该由移动电话来处理。因此，目前的难题是如何实现高层次的自动化，Web 服务技术为这一问题提供了一个初始答案。

8.3 Web服务

为了更好地支持用户在服务中的交互，移动电话需要创建相应的客户端对交互进行管理。实现一个针对某服务的客户端首先需要有关于该服务交互协议的明确规范。在规范中，应明确规定消息格式、发送消息的目的端口和接收消息的源端口等内容。针对上述核心内容，Web服务技术提供了一系列语言和规范，用以支持具有平台无关性的客户端进行自动编译。

8.3.1 Web服务描述语言

Web服务描述语言（WSDL，参见 http://www.w3.org/2002/ws/desc/）是Web服务技术的基础，它用于规范服务执行的操作和调用这些操作所需的消息。WSDL文档结构如图8-1所示，其理念是每个服务都是一个接口，并对外展示该服务执行的操作。每个操作则详细说明从输入到输出的转换过程（除了输入/输出操作，WSDL还支持多种其他操作，比如input only操作只接收信息，并不向客户端反馈任何消息，而output/input、output only等操作则将相关信息发送给客户端）。操作中的输入、输出参数在消息规范中进行定义，其中消息的各部分对应各参数，而这些消息的结构则利用XML Schema定义语言定义成类型规格。为了支持操作的调用，每个操作将与一个绑定关联。绑定中将指明协议的类型，并且每个绑定与一个端口关联，而端口则用于指明消息的源和目的地。

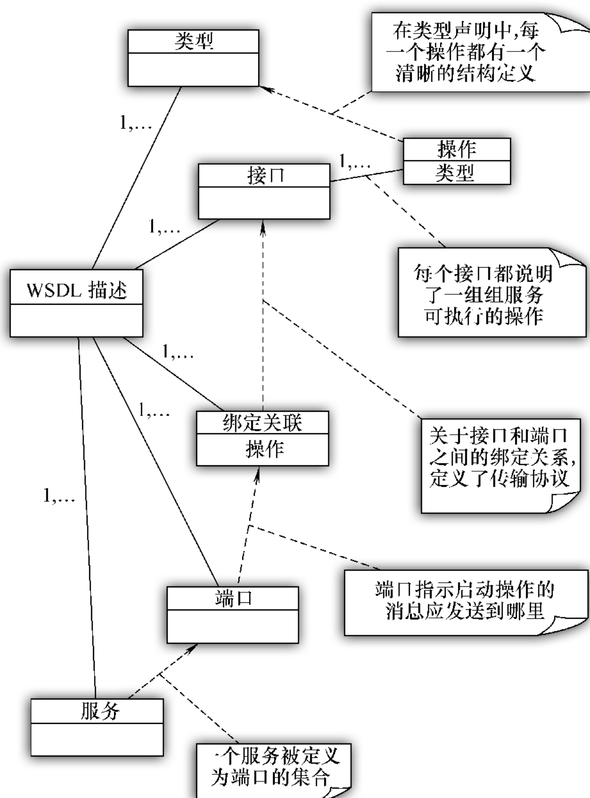


图8-1 WSDL文档结构

由于 WSDL 可用于向服务的潜在用户展示服务的具体功能，那么它就可用于展示意大利和德国铁路系统的时刻表服务。这两个服务都可描述成一个操作，其输入为离开和到达的城市以及离开的日期和时间，输出则为时刻表。消息和绑定部分将指明不同的消息格式以及消息发送的目的地电话号码。

WSDL 有很多优点：首先，它抽象了服务实现细节，只对外展示服务功能；其次，它支持基于同步和异步消息传递的接口的规范化，并允许多种服务类型的规范化。此外，困扰 Corba（参见 http://www.corba.org/）和 DCom（参见 http://www.microsoft.com/com/default.mspx）等分布式计算方式的制约条件（比如平台、语言、操作系统、厂商依赖性）对 WSDL 毫无影响。这些特性使得 WSDL 在移动计算中的应用颇具吸引力，甚至开放移动联盟（OMA）已提出相关指导意见，指导如何在移动计算环境中使用 WSDL[8]。3GPP 也提供了相关规范，以指导 WSDL 与 SMS 及其他协议的结合使用[9]。

WSDL 无疑有助于解决服务与移动平台间数据交互的问题。然而，数据一旦进入智能终端，如何进行处理以及如何选择数据发送给服务，这些问题还有待解决。严格地说，数据使用是在 Web 服务标准（尤其是 WSDL）讨论范围之外的，因为这些标准集中于接口的规范化。然而对于所有使用 Web 服务的应用来说，如何使用数据都是一个至关重要的问题。通常来说有两种方式可以解决这个问题：一是实现定制的代码解决不匹配的问题；二是发掘其他计算机制去提取 WSDL 描述数据的“意思”，并以此为基础来解决问题。在 8.4 节以及后面的章节，我们会讨论如何使用本体技术（比如第 7 章中讨论的 OWL-S）实现上述目的。

8.3.2　统一发现、描述与集成规范

WSDL 有助于服务调用的过程进行，而移动平台还应知道服务的 WSDL 规范所在位置。这一发现过程是借助统一发现、描述与集成规范（Universal Description Discovery Integration，UDDI）[4]完成的。Web 服务应用时需要指明相关的知识库，而 UDDI 则通过定义上述过程所需的一系列相关标准，来支持发现过程的实现。特别指出的是，UDDI 定义了注册和查询相关知识库所需的操作类型。此外，UDDI 数据结构（见图 8-2）还提供了一种表示服务的方法，以便顺利发现服务。在 UDDI 中，服务提供者用“Business Entities”表示，每个提供者会注册多个称为“Business Services”的服务，而每个服务则与一个绑定规范关联。在 UDDI 中，所有实体需符合 TModel 规范。而作为针对服务可假设特性的抽象规范，TModel 可用于多种应用，比如指明服务的 WSDL 描述所在位置[10]，或者基于标准服务分类法（如 UNSPSC，详见 http://www.unspsc.org/）对服务进行分类。

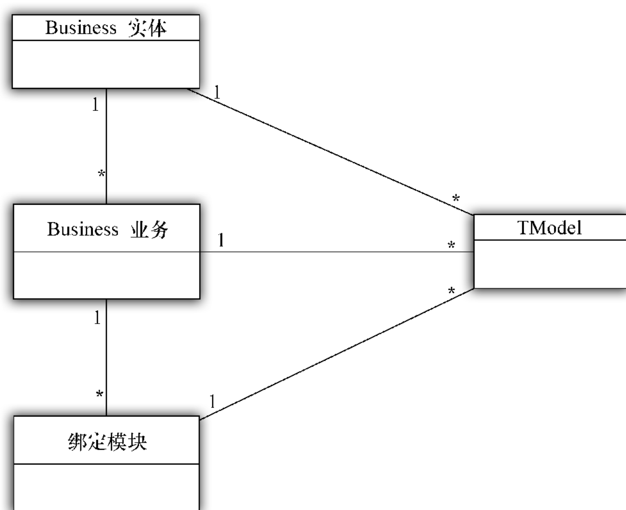


图 8-2　UDDI 数据结构

TModel 表示法功能非常强大，同时也具有很强的通用性。比如说，两个不同的 TModel 可用于表示同一个特征，同样相似的内容可采用不同的 TModel 表示。在理想情况下，工业组织更愿意消除上述多义性，提供特定标准的工业 TModel，但这一工作尚未启动。因此 UDDI 在表示服务的明显特征方面，还难以提供相应指导。

8.3.3　面向服务的体系架构

UDDI 在面向服务的体系架构（Service Oriented Architecture，SOA）中发挥着重要的作用，它负责提供用于 Web 服务发现的中心注册组件。依据 SOA（见图 8-3），服务提供者在注册中心（通常为一个 UDDI 服务器）发布自己的服务信息，然后等待其他服务请求者的发现。服务请求者向注册中心发送请求，提出待查找服务的要求，注册中心则返回符合要求的服务的注册信息，最终请求者选择最合适的提供者，并与其交互。

由于 SOA 模仿 Web 体系架构，在所有服务交互中提供中心化的搜索引擎，其非常适用于分布式部署在互联网上的 Web 服务。然而，搜索引擎的中心化特性在移动计算环境中的应用是个大问题，因为在这种环境中，有些服务可通过互联网在任一地方访问，而有些服务只在受限的区域中能够访问。如果按字面意思去理解，SOA 建议从一个注册中心到另一个中心的移动也就伴随着环境发生了变化。然而很多服务的使用空间范围非常有限，比如基于蓝牙的服务只有几米远的使用范围，基于 RFID 的服务甚至只有几厘米的使用范围。因此上述这些服务

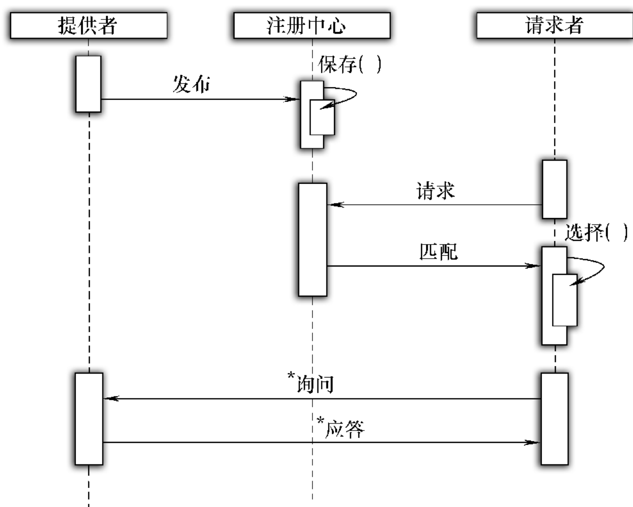


图 8-3 基于 SOA 的交互过程

可能很难访问任何 UDDI 中心。因此，这些服务如果采用和客户端以 P2P 的方式连接并交互，其使用效果会更佳。

WSDL、UDDI 等 Web 服务标准为支持移动平台（比如智能手机和可用的服务）间的交互提供了基础功能。它们也提供了基于 XML 格式的规范，从而降低了服务与客户间互通的壁垒，推动了移动应用间的交互。然而，它们所提出的假设，无论是从表示法角度还是架构角度，对于移动计算来说都过于严格。从表示法角度来说，这些标准只提供了语法互用性，并没有提供数据交换的语义表示法。最终，它们将因为缺乏语义互用性，而阻碍客户与服务间的自动互操作。从架构角度来说，对于注册中心的依赖性将减少在非常有限的地理范围内偶然发现当地服务的可能性。在接下来的章节，我们将尝试分析解决这些局限性的途径。

8.4 从 XML 到本体

在 Web 服务中，XML 是客户与服务间的通信语言，而 Web 服务的问题就在于过度依赖于 XML。由于 XML 建立在定义明确的语法基础之上，因此服务及其客户将用同样的方式解析同一个 XML 文档，从而生成同样的 DOM（文件对象模型）。以图 8-4a 中所示 XML 文档为例，该文档是用 XML 描述的意大利列车时刻表服务。任何 XML 解析器都能够识别该文档的架构，并生成一个等价于图 8-4b 所示的 DOM 模型。

XML 的问题在于一旦其结构确定，很难将文档与客户的其他信息关联。更

重要的是，客户可能会觉得比较麻烦，即认识文档的各个部分，却不理解其中的意思。由于我们理解“time”、“da”、“a”、“Milan”、“Rome”和“date”等词汇，所以我们可以理解到这个文档描述一个已指定时间的从米兰到罗马的旅程。然而在XML文档中，没有将含义与标签关联的机制，因此含义是被程序员编码进接收文档的应用中的。

程序员需要处理数据的含义，这一需求对于泛在和移动计算来说是一个大问题。移动用户快速地变换环境，不断变化需求，但其至关重要的要求仍是准确且最新的信息。他们不可能停在路边，等待所需服务对应的客户端开发完成。相反，移动电话更需要尽可能自主、自动地适应变化的环境，发现新的服务并与其交互。

为了使程序员脱离上述困境，移动设备必须在语法互用的基础上获取接收消息中包含的含义。以图8-4中所示的XML文档为例，移动电话必须理解<to>和<from>两个标签定义了一次旅行的关系以及出发和到达的城市。实际上，移动平台必须能够从接收到的表面信息抽取其中含义，以便理解这些信息如何适用于移动电话可获取信息的通用上下文。

从根本上来说，XML以及XML schemata的问题在于它们只能表示数据的语法，而无法表示其中语义。人们正在尝试使用语义网技术[6]（尤其是OWL[11]，即Web本体语言）来解决这一问题。语义网的目标在于描述本体概念，即指定领域内的概念化，以表示用于刻画用户世界的关系。

```
<queryMessage>
  <da>Milan </da>
  <a>Rome </a>
  <time> 10:30 </time>
  <date> 27/06/2005 </date>
</queryMessage >
```

a)

```
queryMessage.da = Milan
queryMessage.a = Rome
queryMessage.time = 10:30
queryMessage.date = 27/06/2005
```

b)

图8-4　XML文档（图a）及其DOM结构（图b）

对于OWL来说，其特别之处在于区分出两类数据：T-box和A-box，其中T-box用于描述对象和事件类间的抽象关系；而A-box则根据T-box中描述的关系，提供用户可理解的对象描述。例如，T-box可将火车旅行描述为一类具有出发地和目的地的事件，并将火车旅行和其他种类旅行（比如飞行或驾车旅行）清晰地区分开来。此外，它还可以在旅行类型上加以限制：比如飞行需要有出发和到达的机场航站，而火车旅行则需要有出发和到达的火车站点。再者，它还可以指定火车站点和机场航站与实体相关。而在A-box中，则包含对实际火车站点的描述，比如米兰中心车站、机场以及特殊旅程。

另外，OWL还基于描述逻辑[12]提出另外一种逻辑，用于从T-box和A-box

中导出结果。例如，一个具有出发车站和到达车站的对象会被自动划分为火车旅行。此外，描述逻辑推理引擎还能够校验针对服务的陈述的一致性，这样，一个没有列车站点描述的火车旅行会被自动判断为非法。

表面上看，本体只是简单模仿了表现面向对象编程（Object Oriented Programming，OOP）特色的类和实例间的区别，而这点完全可以采用 XML schemata 表示。实际上，XML schemata 允许构建与 OWL 生成的分类结构相同的结构，比如将“travel”设为“train travel”和“air travel”共同的超类。此外，无论描述逻辑还是 OOP 都提供了继承性，比如说，如果“journey”有两个类变量“to”和“from”，那么这些变量同样是其子类可用的。

然而这些相似点是非常表面化的。因为 OOP 并不提供推理功能，所以必须显式指明关于已知对象的所有信息。理想情况下，可以定义一系列在互联网上共享的标准 XML schemata，用于描述通用的一系列旅行种类。后来的一些尝试都是依据这样的思路，比如说 EBXML（http://ebxml.org/）、Rosetta Net（http://www.rosettanet.org）以及更通用化的 EDI 标准。当然，这种方式的问题在于不同组织对于相同的对象可能有不同的看法。比如说，从账务角度考虑，铁路公司可能需要区分工作日和周末的旅行，以便为工程师支付加班费用；另一方面，用户可能需要区分商务旅行和私人旅行，以便用公司信用卡支付前一种旅行，而用个人信用卡支付后者。在上述情况下，同一个旅行说明可能会在服务侧和用户侧采用不同的方式加以分类。如果采用标准的 XML schemata 实现，那么分类信息必须要嵌入编码中；而如果采用 OWL 表述，推理机就会自动处理相同的分类，而无需程序员参与。

OWL 提供的自动分类对于支持移动环境下客户端与服务的互通性至关重要。服务和客户端对应的本体可能大不相同。客户可能更关注不同的交通方式以及各自使用的条件。相反，铁路公司会更关注铁路交通，但是它会有一个很详细的本体去指明铁路旅行涉及的人员和设备。然而只要服务和客户端共享一组通用的概念，它们就可以正确地互通。

最终，移动服务需要将 Web 服务技术和语义网技术相结合。Web 服务标准提供了相关的信息，允许自动生成客户端，并使用正确的交互协议在正确的端口提供信息。另一方面，OWL 则提供了编译这些信息的方法，并使得接收方能恰当地使用这些信息。

8.5　使用语义网技术表示服务

OWL-S[7]通过使用 OWL 本体，定义了如何解释基于 Web 服务标准（如 WSDL和 UDDI）生成的声明，从而在语义网技术和 Web 服务之间建立了连接的

桥梁（此处相关讨论可参见 OWL-S 1.1）。

上层本体提供了一个描述概念或现象的通用概念框架，在此基础上就可以定义其他本体，以提炼或专业化上层本体。例如 MobiOWLS 就用于专业化 OWL-S，以描述移动服务。简而言之，OWL-S 的目的在于为服务提供上层本体，以解答如下问题：

1）服务会提供给客户什么能力？

2）服务如何实现其功能？

3）客户端如何与服务交互？

这 3 个问题与我们在前面提到的 Web 服务面临的 3 个难题紧密相关。第一个问题直接针对服务的发现：服务请求者需要查找具备相应能力、可以达到相应目的的服务。第二个问题是服务组合的基础：当多个服务组合为一个复杂服务时，需要对服务的操作进行细致的描述。第三个问题则是服务调用的主要内容：最终，客户需要知道服务需要什么消息，会返回什么消息，以及这些消息如何与服务的语义表示映射。

为了回答上述问题，OWL-S 分 3 个主要部分对服务进行描述。服务配置部分通过描述服务的能力，来解答第一个问题。过程模型部分通过展示服务工作流回答第二个问题。此部分的主要作用在于支持涉及多次消息交换的交互过程，指明在消息交换时服务需要什么数据，它会上报什么数据，以及更重要的一点，即过程执行的结果。最后，服务绑定（Service Grounding）部分则详细地描述在过程模型中展示的过程如何映射到 WSDL 操作，以及它们如何生成在服务与客户间的具体消息交换。

在 OWL-S 中直接使用 WSDL，体现了 OWL-S 的一个重要设计理念：OWL-S 不是用于取代 Web 服务标准，而是要用语义信息增强其功能的。因此，只要服务的输入输出可以映射到现有的本体，OWL-S 就可以应用在此服务中。实际上，Amazon.com 的 Web 服务[13]已使用 OWL-S 提供相应的描述，并且在应用过程中 Amazon 无需更改 Web 服务的相应代码。总而言之，OWL-S 可用于描述针对移动计算的任何服务。

实际中，定义一个独立于 WSDL 的 OWL 服务绑定是切实可行的。这一点已被 Masuoka 等人[14]证实，在其相关文章中他们定义了 UPnP grounding[15]。然而到目前为止，最通用的方法还是使用 WSDL。

应用语义对于针对移动计算的服务至关重要，由于前面已证实 OWL-S 是支持我们观点的技术基础，我们将在下面对 OWL-S 做一概述，在此过程中我们将提出一个命名为 MobileOWL-S[16]的新应用，这一应用描述了专门用于移动计算的一个专业化的 OWL-S。此外，我们还将列举一些 OWL-S 用于泛在和移动应用中的一些实例。

8.5.1 服务配置

OWL-S 配置文件（以下简称配置文件）用于描述服务的能力。在 OWL-S 中，服务能力是依据服务提供的功能以及一系列附加需求（如安全限制和服务质量）加以区分的。其中关键点在于，能力的定义是和服务提供能力的方式毫不相关的。总的来说就是，能力定义需要描述服务做什么，而不是服务如何实现相应功能。

配置文件的结构如图 8-5 所示。依据上述讨论的观点，配置文件可被分为两个部分。第一部分是服务的功能性能力，来描述服务用于做什么。如图 8-5 左边部分所示，描述这些能力时，既体现了服务实现功能的信息方面内容，也体现了转换相关内容。更准确地说，配置文件中体现的观点是服务提供了信息的转换，即输入被转换为某些输出。例如，一个报告列车时刻表的服务可被描述为这样一个转换，即给出出发和到达城市的名字，则输出两城市间的列车时刻表。一些服务也可能带来物理世界的变化。比如，一个火车订票服务会将出发和到达城市及一个信用卡号码作为输入，并输出电子车票以及票价。此外，服务会在相关信用卡账户中扣除对应车票票款。

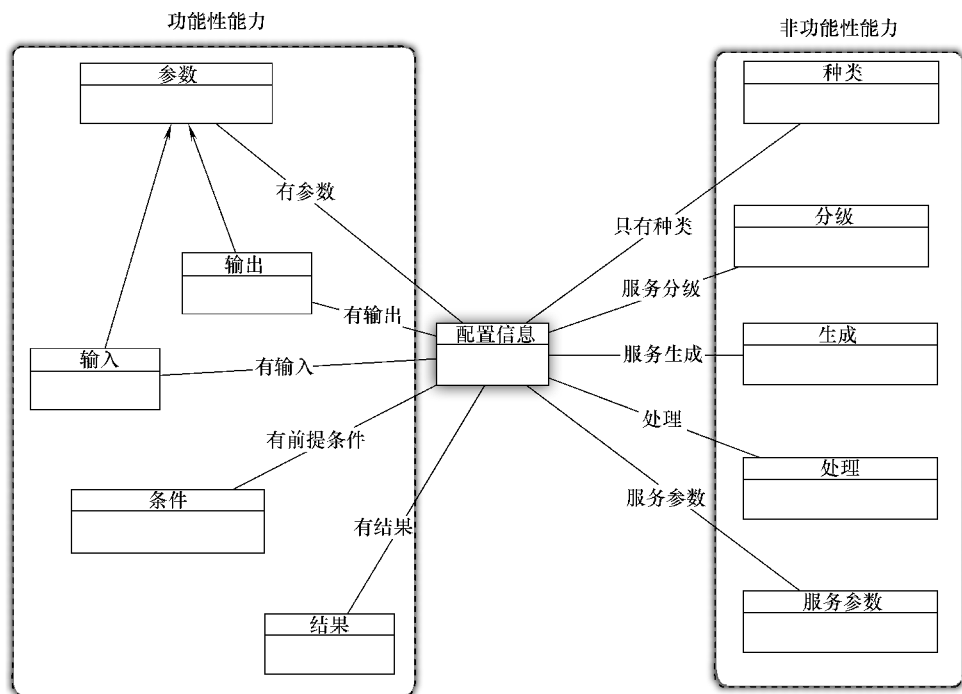


图 8-5 OWL-S 配置的结构

这些功能性能力对于服务的描述是完全不够的，因为两个服务可能会提供同样的功能，但实现的效果却大相径庭。例如，一个可以快速实现功能，而另一个却特别缓慢；一个需要高带宽，另一个不需要；一个接受 Visa 卡支付，而另一个要求 MasterCard。由于用户可能对服务选择有很强的偏好，上述区别对用户来说至关重要。如图 8-5 右半部分所示，服务配置中的非功能性能力可以准确表达这些特点。

非功能性能力是通过服务参数（Service Parameter）来表示的，此部分内容定义了在配置中添加新特性的生成方式。特别是，对应每一个描述服务的特性，都可以添加一个新的服务参数。例如，服务参数可用于指明服务的地理范围，或者使用一些分级图式指明质量级别。此外，服务参数也被用于表示安全能力[17]。总的来说，由于服务参数集理论上是无限的，所以 OWL-S 服务参数的一个优点就在于只要需要描述服务的一个新特性，就可以增加一个新参数。

除了服务参数，OWL-S 还提供了别的表示服务特性的方法。第一种方法是指明服务产品（Service Product），即与服务交互产生结果的类型。例如，一个书籍销售服务会指明其处理产品的主要类型是书籍。此外，配置文件还支持服务分类（Service Classification）和服务类别（Service Category）的规范，这些内容用于指明如何基于服务词典（如 UNSPSC 或基于 OWL 的本体）对服务进行分类。如图 8-6 所示，一个词典可表示出服务的不同类型，比如商务、通信、娱乐、信息和旅游。这样，就可以基于上述词典将所有服务划入对应的类别。

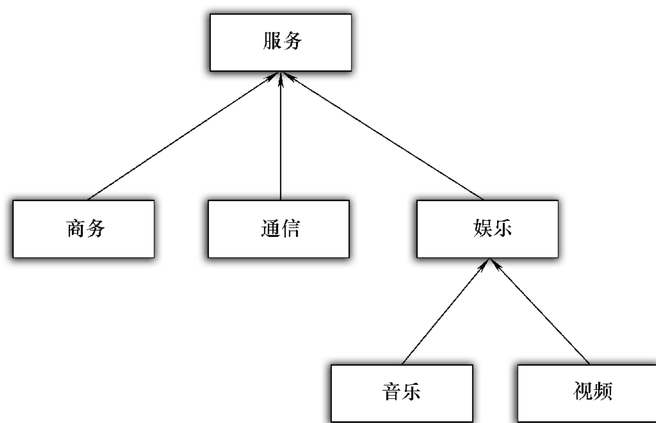


图 8-6　服务本体的实例

1. 移动性特征的表示

OWL-S 配置允许依据信息层面和物理层面的转换对服务能力进行规范化，但对服务参数却完全不做规定，而把这个问题留给 OWL-S 的领域去专业化。在

本节中，我们打算针对 OWL-S 配置定义一个新类，专用于移动计算领域。我们把这一专业化成果叫做 MobilOWL-S。我们认为有一些新的参数对于移动计算领域至关重要，并在 MobilOWL-S[16] 扩展了原有的 OWL-S 定义，加入了这些参数的详细规范。MobilOWL-S 的扩展主要体现在如下 4 个特性中：

1）Comm_ Channel：这是一个用于在服务和移动平台间传输内容的通信信道。使用该特性就可以指明服务是否使用了蓝牙、SMS 或其他协议，以及使用相关协议在范围和带宽上的限制。比如蓝牙服务只能在几米远的受限范围内使用。

2）Cost_ Model：这个成本模型用于服务的计费。它可以是统一费率、依次收费或者免费。

3）Media：这是一个用于向平台传递信息的媒体类型。例如，媒体类型可能为普通文本、声音或视频。

4）Device_ Requirements：服务可能会针对信息的显示设备提出一些需求。这些需求用于指明期望的屏幕尺寸、分辨率、内存、处理速度及其他类似的参数。

MobilOWL-S 针对移动服务的服务参数提供了高层的描述。例如，娱乐服务的规范需要指明显示内容类型的等级以及期望的用户类型，而位置服务则需明确定位方法，因为不同的定位方法的精度截然不同，特殊的服务可以通过实例化 MobilOWL-S 类生成。

2. 使用配置发现服务

服务发现的过程涉及两方：服务提供者一方描述自己的服务，而服务请求者一方则需要提供待查找“理想”服务的描述信息。服务发现的问题在于判断提供者的服务是否符合请求者所需服务的描述，由于服务请求者并不知道有哪一类服务是可用的，同时也不清楚服务是如何描述的，所以服务发现是个有难度的过程。因此，即便服务的请求信息和发布信息描述的服务非常相近，它们的内容也会大相径庭。由于上述问题，使用字符串匹配的发现算法很难识别两个服务描述的相似性。实际上，匹配算法应抽象于服务的表面描述，以便当两个服务语义相同时能被及时识别。

OWL-S 通过使用本体及其下层逻辑，可以提供一种实现抽象过程的方法。因为本体和逻辑推理的应用可以使匹配引擎检验出发布服务的描述是否等价于被请求服务的描述，并且这一校验过程完全独立于服务描述的表面格式。

在近几年，业界开发出不少用于 OWL-S 的 Web 服务发现算法，其中包括 OWLS-UDDI 匹配器[18,19]、RACER[20]、SDS[21]、MAMA[22]、HotBlu[23] 和 OWLS-MX P2P 查找[24] 等。虽然它们在技术细节上并不相同，但都提供了重点实现输入输出匹配的匹配过程。从本质上来说，这些算法用于确保服务请求中的输出与提供服务的输出完全匹配，并且服务请求中的输入也与提供服务的输入匹配。

在匹配过程中使用的基本推理类型如图 8-7 所示。该图中使用了与图 8-6 相

同的本体。假设请求者在查找可以提供娱乐的服务，这时查询引擎就会开始查找与请求匹配的服务。另外，它也会同时查找在词典中更高层次概念描述的服务（如Service）或者更低层概念描述的服务（如Music或Video）。在更高层次定义的服务比需要的服务更概括，只要它们的定义中能（或不完全能）提供需要的所有信息，它们就会近似于期望的服务。而更低层次定义的服务则更加具体，因此它们所提供的服务与所需服务严格近似。

上述描述算法提供的匹配过程只能进行输入输出的匹配，某种程度上说，只是服务参数的匹配。在未来的发展中，发现算法需要解决在发现过程中其他方面的问题，比如服务描述前置状态和结果的匹配，实现了这一点，请求者在使用服务后，就可以指定想要的条件了。

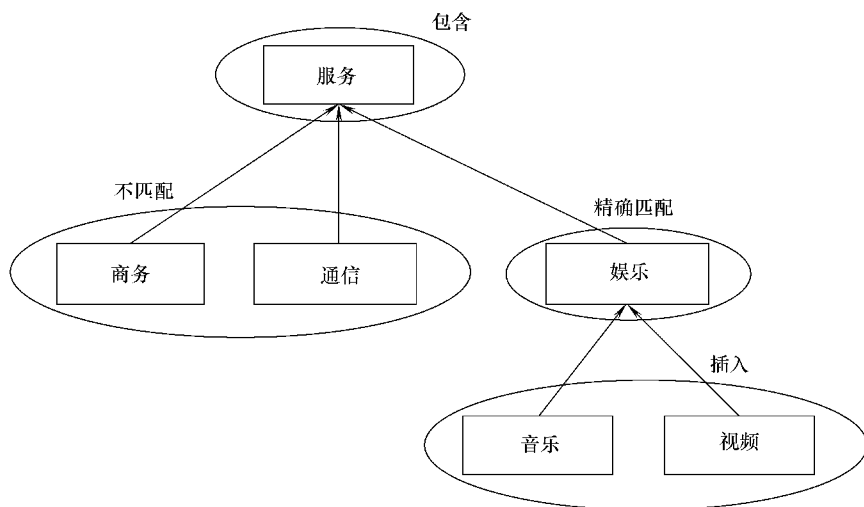


图8-7　本体和匹配度的实例

8.5.2　基础设施无关性

OWL-S中发现过程的一个重要特性是OWL-S并不会像SOA和UDDI那样做出架构上的承诺。服务请求者和发布者之间的匹配过程是一个可实现的功能，并且可以应用于区别很大的基础设施的上下文中。实际上，OWL-S已被应用于完全不同的基础设施中，比如没有任何集中式注册中心的理想P2P环境，以及具有管理发现和交互过程的集中式代理或UDDI的环境。从理论上来说，OWL-S甚至可被应用于本书讨论的任一基础设施之上。

不同的发现架构在效率和可用性方面带来完全不同的权衡方案。例如，中心代理允许非常轻量级的客户端参与交互，因为它将许多可分布式部署于客户端的

功能集中在一起。这一点同时也带来了产生瓶颈的可能，而这一瓶颈很可能导致单点失效。理想的 P2P 架构则假设服务和客户端不仅仅需要管理它们之间的交互，还要管理相关的发现过程。但这一假设使得基础设施异常复杂和多变。最终，应用提供者将通过对基础设施和应用模式的评价，选择最合适的基础设施。而无论结果如何，OWL-S 都可以应用其中。

8.6　过程模型和绑定

发现过程用于找到满足用户需求的服务，但是它完全忽略了请求者和找到服务之间的交互。两者间的交互是通过使用过程模型和绑定指明的。其中过程模型指明组成服务的过程以及服务所需接收信息的发送顺序；而绑定则用于表示过程规范与对应操作的 WSDL 规范间的映射关系。

更准确地说，过程模型被规定为一组过程，其中一些过程被命名为组合过程，用于指明控制流的声明，另一些则被称作原子过程，用于说明用户和服务间的各个消息传递。OWL-S 提供了许多类控制流的声明，比如用于指明并发过程的顺序和非顺序、分支以及合并；用于指明不同进程间非确定性选择的选择；用于指明不同进程选择标准的条件选择等。此外，OWL-S 也允许使用许多循环结构体表示过程的重复性。

图 8-8 是一个复杂过程的实例，它展示了简化版的亚马逊（Amazon）Web 服务的过程模型。这个过程模型提供了 3 个选择项。第一个是浏览产品，这一选择将需要决定浏览哪类产品。第二个选择项是管理用户的购物车，这一选择也需要用户决定如何修改购物车。第三个选择项则是购物，这一过程是作为前两个操作的结果进行实现的：首先用户浏览亚马逊的数据库，找到她所需的产品，然后将产品加入到购物车中。

除了指明控制结构体，OWL-S 的过程模型还指明了其中的数据流。数据流主要描述有关数据如何在过程间传递的规范。例如，在上述亚马逊的过程模型中，在选择操作中选取的产品信息必须传递给账务管理操作中，这样产品才能加入到购物车中。本质上而言，数据流技术可以允许服务发布客户端需要保存什么状态信息以及何时使用这些信息。

绑定则用于指明原子过程如何映射到 WSDL 操作，以及最终原子过程如何完成调用。OWL-S 和 WSDL 之间基本的映射关系如图 8-9 所示：原子过程映射为 WSDL 操作，OWL-S 输入输出消息映射为 WSDL 输入输出消息。后一个映射尤其重要，因为它指明了消息的显式语义。如前所述，使用 XML 模式（schemata），而不使用本体，将在自动调用过程中阻碍 WSDL 的使用。OWL-S 的映射则会解决这一问题。OWL-S 的映射将指明 WSDL 使用的 XML 模式（schemata）如何映射到

OWL本体中定义的概念，从而有效地解释了服务与客户端间传递的数据。

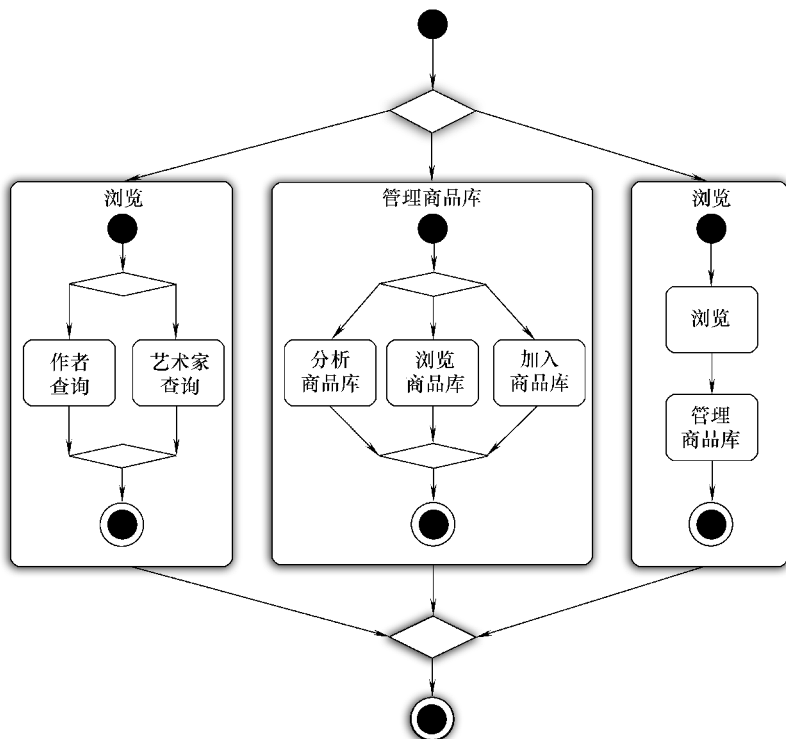


图 8-8　Amazon Web 服务的过程模型

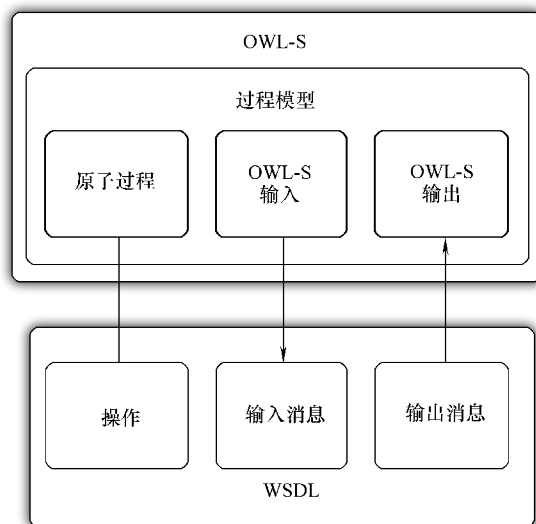


图 8-9　OWL-S/WSDL 映射概览

8.6.1 使用 OWL-S 过程模型与服务交互

OWL-S 和 WSDL 的结合为控制移动应用与服务间的交互过程提供了强大的支持。OWL-S 从多个方面丰富了 WSDL。首先，它为消息的规范化提供了显式的语义，这样服务与其客户端间的交互信息就能够以双方都能理解的方式进行规范。第二，OWL-S 指明了 WSDL 操作执行的顺序。例如，图 8-8 中的过程模型包含 7 个原子过程，并分别对应 7 个 WSDL 操作，但是 WSDL 文件对接下来执行哪个操作毫无帮助。实际上，OWL-S 指明了操作执行的顺序，以及选定一个指定操作的结果。举例来说，它将指明浏览书籍目录的结果是书籍的选择，接下来这一信息会有助于客户决定下一步要执行的最佳操作。如果客户想购买 CD，那么浏览书籍就会毫无用处。

此外，OWL-S 过程模型也会突出说明在实现与服务的自动交互时用户需要满足的需求。第一个需求是执行必要逻辑推理的能力，推理过程将使得客户可以解释其从服务接收的信息。第二个需求是客户能够掌控服务的过程模型，以达到预期结果。本质上来说，就是客户需要对预期目标有一个显式的描述，并且客户需要能够将此描述与过程模型的输入、输出、前提条件和结果关联在一起，以取得预期效果。

8.6.2 使用 OWL-S 查找服务并与服务交互

移动应用首先需要完成的任务是与服务的交互。例如，移动应用可能会对已经身在机场，但想更换航班的用户有所帮助。假使用户需要从慕尼黑飞往米兰，那么移动应用就会将目标设为找到航班时刻表，并购买合适的机票。

目标确定后，移动应用就需要查找一个能满足该目标的服务。查找服务的第一步就是编辑一个能实现该目标的服务的抽象描述。这个抽象的描述使用配置文件（Profile）的形式编辑，在其中移动应用应指明期望的输出以及需要提供的输入。例如，移动应用应指明需要一个服务，该服务需要输出意大利和德国间的航班时刻表。查找过程的第二步是将期望服务的抽象描述与现有服务的描述相匹配。最后，一旦找到所有的时刻表服务，移动应用就可以选择一个或多个服务进行交互。

作为发现过程的结果，移动应用将确定需要调用的服务。下一步就是管理该服务的调用过程，这需要移动应用分析服务完整的过程模型，确定需要执行的过程以及执行的顺序。这一步骤中，移动应用需要对交互过程中产生的权利义务关系进行推理，以保证应用不会在确定用户是否饥饿前就购买食物。确认权利义务关系是个十分必要的过程，只有这样移动应用才能同时与多个服务交互，以便在确定车次或旅行日前选出可能的最佳条件。此外，这一步会提供与用户交互的正

常环境，用户可能已决定不再旅行，或者选用飞行方式。

最终，移动应用会使用OWL-S绑定编辑发给服务的消息以及解释收到的消息。在8.2节描述的与时刻表服务交互的实例中，应用会完成拨打电话和格式化SMS消息的功能，使用户免于这些细节操作。此外，应用还要收集用于和其他服务交互的数据，完成购买车票以及预订旅馆等操作。

8.7 关于Web服务语义的其他方案

本章中我们已指出OWL-S使用显式语义信息来丰富Web服务描述的一个方案。作为第一个针对语义Web服务的方案，虽然OWL-S已经被最为广泛理解的，近年来仍有其他方案出现。尽管本章的目的不是概述所有方案的Web服务描述语言，但为了保障完整性，我们将在此简单讨论WSMO、WSDL-S和SAWSDL等另外3个方案，重点分析它们对于在Web服务中加入语义这一工作的贡献。此外，还有一些相关的有益建议，比如OWL-P[26]加强了定义各方通信策略的重要性，而SESMA[27]则提供了语义Web服务的XML语法，使其可直接映射到Web服务标准。

8.7.1 Web服务建模本体

WSMO（Web Service Modelling Ontolog，Web服务建模本体）（www.wsmo.org）和IRS Ⅲ[28,29]也有和OWL-S十分类似的愿景和目标。它们都是针对如何用语义网中的本体来表示Web服务，并且目的都是使Web服务与客户间能更有效地发现和互操作。然而WSMO采用了不同的途径：它并没有强调把客户侧的推理能力作为互操作的关键点，而是强调了调解作用，以支持Web服务间的自动互操作。

当两个Web服务或某个Web服务需要和其客户互通时，WSMO的调解器将解决不可避免的不匹配现象。这些不匹配来源于不同本体或交互协议的使用，以及两个程序在设计时就针对不同的目标等客观事实。

为了解决互操作问题，调解器应能提供针对重用性和可扩展性两个重要问题的解决方案。调解器有效地提高了可用性，因为通过定义一组用于解决特殊上下文环境下互操作问题的调解器，就可以使统一服务用于不同的上下文中。而减少所需定义的映射关系则会增强可扩展性。实际上，各参与方的一对一直接映射数目将随参与方数目的增加呈二次方增长。这确实是一个严重的问题，因为这需要大量的实现工作，因此倒不如把减少实现需求的希望寄托到定义调解器间的映射上。

调解器的思路的确很重要，但也有问题待解决。在WSMO中一个未明确的问题是调解器的来源。进一步说就是，调解器到底是以Bouquet等[30]提出的逻

辑推理形式出现，还是需要显式实现，再或者是两种方式兼而有之。当然，移动计算的问题就是实现工作需要由忙于赶车的移动用户完成，这基本等同于技术失败。

到目前为止，WSMO 更像是要推动 B2B 交易，而不是支持移动用户。其实这一点并不奇怪，因为 WSMO 建议初期思想的提出就是基于 B2B 计算和企业数据交换[31]的思想。不过现在也有一些尝试，他们基于 WSMO 上下文提出，旨在解决移动用户的问题。

8.7.2　WSDL-S 和 SAWSDL

WSDL-S[32]尝试在 WSDL 中补充语义信息，并希望这一举动能够方便 Web 服务的发现和互操作，以利于商业集成。WSDL-S 的主旨在于提供一个实用的方式为 WSDL 补充描述能被程序员直接使用的语义信息。通过在 WSDL 中使用“可扩展元素”加入语义，其中“可扩展元素”是一种用于在 WSDL 中添加信息的机制，但该机制无需在 WSDL 规范中指明相关内容。特别是，WSDL-S 允许对用于描述数据传输格式的类型进行语义规范，并且允许数据格式和语义规范之间的映射，这使得在数据交换中抽取语义成为可能。此外，WSDL-S 还允许对操作的前提条件和效果加以说明，并且增加了一个元素，以支持对服务类型的规格化。

同 OWL-S 相比，WSDL-S 提供了类似于绑定的规范。据 WSDL-S 的制定者所言，这一方式比 OWL-S 绑定有更多的优势。第一点是在 WSDL 中直接可用语义，而不是像在 OWL-S 中，语义是在与 WSDL 操作关联的原子过程中指明。第二点是它提供了一种途径，使语义可用多种方式进行表达，而这一点 OWL 无法企及。理想情况下，开发者也可以使用 UML 指明语义，这样就减少了对显式语义的需求。

WSDL-S 正在逐步发展成为 SAWSDL（http：//www. w3. org/2002/ws/sawsdl/）。SAWSDL 是一种用于直接对 WSDL 文件标示语义的语言，它已作为 Web 技术的标准化主体，成为 W3C 的建议。由于 SAWSDL 已进入标准化状态，其他 Web 服务技术都在尝试使用其内容。例如，现在已经有人尝试研究如何将 WSMO[33]和 OWL-S[34]的基础内容引入到 SAWSDL，以挖掘语义标识。SAWSDL 的主要目标是提供一个非常轻量级的 WSDL 语义标识，以提高工业界的接纳度。在此范围内，SAWSDL 并不直接支持一些已被所有语义 Web 服务语言采纳的选项，如前提条件和效果（尽管 SAWSDL 可能采用间接方式将前提条件和效果与操作关联）。SAWSDL 的一个问题在于简化表示法会带来精确度的降低[35]，并且在 OWL-S 中针对一个绑定需要提出许多关于标识的假设，以解决标识的多义性。不过，在被推往标准的同时，SAWSDL 已成为工业界早期采纳者的第一选

择，而且它对于在移动计算上下文环境的实际采纳来说已足够轻量级。其他标准化的努力将依赖于SAWSDL的推进以及其使用时带来的问题。

8.8 语义在Web服务中的应用

在上一章节中，我们已提出一种服务表示方法，用以解决移动应用面临的问题，即需要同服务进行自动的交互。在本章节中，我们将举例说明一些包含采用语义描述的服务的具体应用。

8.8.1 使用语义服务方便用户的交互

通常情况下，在用户请求服务与现实环境可用服务之间存在一定的差异。这一差异的一般解决方式是通过在设备中硬编码进行相应的设置，使设备能够利用环境，但这一费力的做法需要只有少数用户具备的专门技术，并且通常会造成计算环境的非最优使用。由此出现了一种新的计算范例——任务计算[14]，其目的是通过将所有服务及信息源以语义Web服务的形式展现，以消除用户与服务之间的差异点。这种方式下，就可实现所有服务的自动发现以及它们之间交互的自动管理。

任务计算是由一组工具完成的，而这组工具组成了任务计算环境（Task Computering Environment，TCE）。TCE中最重要的3个工具是STEER、TCI和White Hole。STEER是一个客户端，负责将一组可用服务展示给平常用户。它建立在通用即插即用（UpnP）发现技术之上，使得其可发现指定环境中所有可用的服务。但由于用户环境可能不同于计算环境，STEER去除了对用户无直接用处的所有服务，大大减少了可用服务的数量。STEER的另一个作用在于它针对用户通常遇到的问题提供了解决方案，使得用户无需处理这些问题。

环境中可用的服务是通过OWL-S和WSDL描述展示给用户的。WSDL为服务增添了相应能力，以建立用户设备与服务间的连接，而OWL-S则提供了对服务与设备间数据交换的解释方法。进一步说，OWL-S通过定位可利用其他服务信息的服务位置，提供了构建服务组合的方式，并展示给用户。例如，当检测到某一服务提供了某人的联系信息时，STEER就可以使用此信息的所有服务，并基于两个服务生成一个组合服务。通过使用这个组合服务，用户就可以发现朋友的地址，并看到通往该地址的路线图。

任务计算使用到了语义的精华部分，特别是OWL-S。OWL-S用于支持后绑定，而不是预先配置在指定环境中工作的设备。这样设备就能在相应环境中正常工作。此外，OWL-S被用于支持服务组合，这样在指定环境中可用的能力就能正常展示给用户。

8.8.2　使用用户上下文和偏好进行计算

由于上下文会影响用户问题的解决方案，任务计算提出了考虑用户操作上下文带来的问题。任务计算提出的第二个问题是考虑用户的偏好，以便找出更能满足用户需求的解决方案。上下文和用户偏好的表示都是泛在计算和移动计算领域积极研究的问题。而在这些领域，语义信息的表示和逻辑推理的应用发挥着重要作用。在第 7 章更为详细地讨论了上下文推理和本体，本处讨论这些则是为了保证内容的完整性。

使用上下文的计算需要显式表示用户上下文的定义。泛在应用的标准本体 SOUPA[36]是一个 OWL 定义，它尝试定义字面上已定义的上下文的不同方面。特别是，SOUPA 包括借助朋友的朋友本体、空间关系和策略规范等实现时间和任务关系定义的子本体，从而允许对资源使用限制及其他内容的规范化。

SOUPA 被用于实现一个称之为上下文代理架构（CoBrA）[37]的泛在计算环境，此计算环境主要为移动用户使用智能房间提供支持。CoBrA 包括一个上下文知识库。这个知识库将提供上下文信息的一致性存储，既包括例如同一用户不能同时在两个房间的一般限制条件，也包括记录特定时间房间中的人物信息。上下文获知模块增强了上下文知识库的功能，以便利用传感器收集房间的当前状态信息。而这些信息将用于上下文推理器。推理器推理出一些隐式信息，比如房间中人物的角色以及传感器无法检测的人物行为类型。最后，CoBrA 使用策略管理模块保证房间内的整体策略与用户的个人策略都得到满足且互不冲突。

OWL-SF[38]也同样强调了使用语义表示和使用上下文信息的需求，它采用一种上下文的 OWL 表示及底层的推理机制来推理房间的状态，并实施策略。在 OWL-SF、CoBrA 以及其他类似于 My Campus[39]的其他项目中，最突出的一点是传感器用于采集上下文信息，而覆盖传感器的是大量异构和分布式的信息源。语义 Web 服务模式充分利用了语义表示下的推理功能，有助于提供信息源的描述，使信息源对用户来说快速可用。上下文信息的收集和使用同样也带来了一些问题，比如说如何选择能够为用户提供最合适信息的服务，这也是发现过程中经常被遗忘的内容，也就是在功能相同但服务质量不同的服务中进行选择。MobiOnt/MobiXPL[40]为解决该问题提供了一个初步解答，即使用本体定义一组反映用户需求的偏好关系。此外，MobiOnt 还提供了一个计算方法，用于挖掘偏好关系，以决定哪一个服务最能满足用户需求。

尽管 MobiOnt 为发现过程提供了初步方案，但仍有一些主要的难题尚待解决，比如说上下文信息和用户偏好如何结合，以便于用户能够表达在某一特定上下文中具有的一组偏好，而这些偏好在另一种上下文中将会有所不同。

8.8.3 使用语义控制能量消耗

自动网络化系统（Autonomic Network System，ANS）[41]模式尝试解决泛在计算中的一个核心问题，即能量需求。泛在环境中的计算通常涉及大量的消息传递，然而，一个设备可能无法提供足够的能量来保持与网络的连接，从而实现消息的接收和发送。

基于ANS的理念，一个面向服务的体系架构应该提供对服务发现的支持，这样用户就可以找到满足需求的服务。执行发现的能力为ANS中服务的自愈性特点提供了基础的功能块，这样在ANS中当发现服务不再可用时，设备会自动查找一个功能类似的服务。

自愈性实现一般伴随着高连接代价，因为客户需要对能实现所需功能的所有服务进行状态监控。ANS假设每个服务都有一个指明服务作用的核心功能，以及需要指明核心功能附加特性的辅助功能。辅助功能包括加解密以及如消息安全、维护服务选择过程中相关承诺的能力等更概括的一些功能。

核心功能与辅助功能的划分带来两方面的好处。第一，它有利于代码的重用，在某种意义上，同一个辅助功能附属于核心功能千差万别的不同服务。第二，辅助功能，特别是承诺方面功能的采用使得服务能够控制其交换的消息数。承诺用于指明服务与其客户间建立的连接以及相关的服务质量。依靠承诺的保障，客户不用不停地查找新的服务，完全可以集中精力做好与某一服务的交互。

支持过程发现的OWL-S，概括一点也就是语义计算可以支持自愈性，因为它们使得设备能够表达所需的能力以及匹配过程。在匹配过程中将实现软件性质的匹配，从而找到能够基本满足设备需求的服务。此外，OWL-S Process Model还可用于描述对应不同辅助服务的工作流片段。在这种方式下，就可以通过合并核心服务以及所有辅助服务的描述，形成一个服务的完整描述。

8.9 问题和未来的挑战

上一节，我们已提出了移动计算的一个愿景，即移动应用需要自动实现与服务的交互，而无需程序员参与客户端代码的实现。在描述过程中，我们突出了Web服务标准带来的贡献以及它们的瑕疵。此外，我们还尝试提出了一个针对显式语义的实例，以允许移动应用与服务间的交互。我们介绍了服务描述语言OWL-S，该语言拓展了Web服务标准，提供了显式的语义。尽管OWL-S以及其他方案（比如WSMO、WSDL-S、SAWSDL等）的研究已取得了很大进展，但仍有许多重要的挑战尚待解决。在这个总结部分中，我们将回顾在移动计算中构建

完整服务技术需要面临的问题和挑战。

8.9.1　循环中的用户

移动计算正在一个矛盾的状态下发展：一方面用户想要控制电话的能力以及它们所做的决定，而另一方面用户又想摆脱技术细节的约束，比如调用服务、委托终端决定如何传输信息甚至传输什么样的信息。

现在面临的挑战就是要找出一种表示服务与应用间交互的途径，同时使得用户能够理解，并能控制其移动电话的活动。这是一个意义重大的挑战，因为这样就可以清晰地区分可从用户处索取的信息和机器中已有且可以直接传输的信息。例如，在每次支付时都要求用户输入信用卡号，用户会感觉麻烦，而另一方面，在用户毫无感知的情况下发送信息卡号将更令用户担心，甚至最终造成用户放弃使用该技术。

进一步说，如果移动电话做出一个决定，它应该能够解释做此决定的原因。移动电话可以询问用户她是否愿意为某次支付使用信用卡，但它也应该解释支付的所有细节，说明那些可能在与服务交互期间清晰可见，但对用户而言却从未披露的账目信息。一般来说，用户希望知道机器在做什么。

8.9.2　信任和隐私

人们通常认为信任和隐私问题是网络层次的问题，但在服务层面同样存在这样的问题。无论何时，只要我们从某处搜集信息，都需要对信息源的可信度进行评估，这一问题在与服务的交互过程中同样存在。

信任有两个层面的意思：一是用户需要信任移动应用选取的服务在以正确的方式处理事情；二是用户需要信任服务不会滥用用户提供的信息。例如，当购买火车票时，用户需要相信服务在购买有效的车票，而不是一张错误火车系统的车票。此外，用户要被保证服务不会未经授权，就去使用用户提供的信用卡信息。最后，用户会需要服务的这样一个保证，即用户提供的信用卡号等个人信息不会在其不知情的情况下与第三方交换被发布在公共网站。

当然，这仅仅是在技术层面的问题。在技术层面上，需要开发相应机制去评估服务和信息源的可信度，需要针对不同方面的信任评估提出完整的理论。然而，在社会层面同样存在隐私和信任问题，这就要求指定相关标准以及法律的相关条例，以保护用户完成交易。

8.9.3　完成循环

尽管针对 Web 服务发现、交互和组合已有很多方案提出，但现在仍没有一个统一的架构。众多不同的工作表明理论上所有的交互步骤都是可以实现的，但

还没人提出针对完整交互过程的方案，这一过程从用户或应用的目标开始，发现所有的可用服务，选择最佳服务，与服务进行交互，并使用交互结果解决原始问题。

目前存在的挑战是既没有技术解决方案去消除期望服务与找到的服务之间的不匹配，也没有技术方案去解决如何使用不匹配的相关信息去通知交互过程需要向服务传递什么信息。然而，这一研究领域还处于研究的初期和活跃期，本体在几年前还仅仅是个梦想，但每次会议都会出现不少新的方案，其进步还是非常快的。

最后一个方面是能产生交互。对于客户和服务正在做什么，一样需要监督控制。需要监控的原因是客户端可能需要向用户解释交易的状态；此外，客户端也可能需要监控服务，以决定何时以及是否需要终端交易[42]。例如，在购买车票时，客户端也许需要知道为什么购买过程的时间比通常情况下要长，为什么会出现意外错误。在Web服务领域中，针对管理和监控交互方面的内容已有大量标准出现，并且近期的成果已引入了语义网技术。

8.9.4　使用上下文信息

对于移动计算来说，另一个问题是相关信息并不是存储在数据库的可靠信息，而是一些含糊不清且不断变化的上下文信息，一般在用户的移动过程中被采集。当在机场选择最近的食品超市时，移动应用需要知道用户的确切位置。但是，可能无法及时获取用户的位置信息；将GPS信息映射到特定终端可能非常困难，因为在封闭环境中GPS很可能是不可用的，而其他获取上下文信息的手段或许并不可靠。

移动计算的另一个挑战是需要将服务的使用与实际可用的信息相关联。如果位置信息不可用，但在与服务的交互中却需要相关信息，这时移动应用就需要能够找到另一种实现目标的方法。

8.9.5　Web服务组合

本章中我们假设存在一个服务可以解决用户的所有问题。虽然在实际中很难提出这样的一个解决方案，而且也是完全不可能的。通常情况下，我们可以想像没有单个服务可以达到用户的目标，然而多个服务的组合有可能提供用户所需的服务。例如，为了实现一个在机场预订旅馆的服务，移动电话可能首先需要确认用户是否有进餐的时间，然后还需要确认飞机是否会晚点以及其他事情。这种情况下，没有单个服务可以实现上述目的，电话就需要将旅馆预订服务、飞机出发服务、用户行程以及其他可能的服务合理地编排在一起。

自动Web服务组合是Web服务研究中的“圣杯”。人们提出很多相关的

Web服务标准来尝试解决组合的问题，这些标准主要采用生成基于XML语言(如BPEL4WS[44]和WS Choreography[45])的方式实现，而这些语言用于指明完整的服务组合过程。这些语言使得程序员可以指明何时服务客户端可以向服务发送消息，以及服务将如何交换信息。然而，这些服务描述的交互是经过仔细编排的。实际上，它们也被叫做编排语言，在这组已用服务中的任何变动，或任一服务交互协议的变化都会导致整个服务组合过程的完全变化，而这种变化的实现是很难自动完成的。

此外，还有一些研究工作尝试编译一个自动组合[46]，这种情况下通常需要一个自动规划器，使用means-end分析[47]导出在特定时间需要什么服务。自动组合的问题在于如何将其与发现过程结合这一问题目前尚不清晰。自动规划器通常先定义一组完备的操作符，然后将操作符组合在一起以解决某个特定问题。当应用于服务组合场景时，首先假设每个服务类似于一个操作符，但此时会遇到一个问题，即服务实现结果的描述与规划期需要结果的描述可能无法一一对应。那么接下来所做的调和过程就需要实现消耗极大计算资源的推理功能。

移动计算的另一个问题是由于用户移动带来环境变化，以及服务经常动态出现或消失，用户的目标也在动态变化。提前设计一个完整计划是毫无可能的，因为在用户遇见的不同环境中会有什么服务可用是完全不可知的。此外，规划要能够区分需在特定条件下完成的暂时目标和用户尝试在其生活中实现的全局目标，并且前一种目标的实现不会阻碍后者的实现。例如，用户在换机时可能需要就餐，然而如果用餐时间过长会使其错过下一个航班，她就可能会选择不就餐而直接出发。用户目标的满意度是可以为实现更重要的目标而牺牲的。

8.10 小结

移动计算的下一个新领域是为用户提供泛在服务。然而问题是移动用户需要一个用来使用服务的交互机制，并且此机制不依赖于客户端的输入。实际上，用户需要一旦发现服务，客户端就会在线实例化，而交互过程将平滑自动地进行，且不需要太多的人为干预。

这一领域为我们提出很多技术问题。其中一点就是服务的数据结构不足以完成相关工作。我们应该提出一种表达消息意义的新方法，使得移动平台能够根据这些信息进行推理，决定需要发送什么信息，以及哪些信息内容来自用户。

Web服务标准提供了描述移动平台与服务之间交互的一种方法，但是它们无法描述传递消息的语义，最终Web服务标准需要程序员对客户端与服务之间的通信进行编码。OWL-S针对这些问题提出了解决途径，但仍有很多问题阻碍

挖掘 OWL-S 描述中本质的所有能力。

然而最大的挑战是服务技术需要解决用户的问题，而且此技术提供的最新和恰当的信息能使用户受益，同时用户仍保持对传输信息和接收方的控制权。最终而言，用户是决定服务技术是否成功的关键，成功与否并不决定于我们可证明的定理，也不决定于我们可创建的原型系统，而是用户从中能够获取的益处。

第 9 章　动态适配——实时调整服务

Robert Hirschfeld

9.1　导言

我们希望 B3G 的移动通信系统不但可以集成多个网络，还可以通过第三方服务的提供大大促进服务的丰富性。在此情况下，要充分解决 B3G 环境下移动多媒体服务的高要求，必须实现对复杂动态计算环境以及在多个服务平台的分布式部署的全面支持。新兴服务和应用的难预测性和复杂性必然导致动态服务适配（Dynamic Service Adaptation，DSA）和不可预期的软件演进（Unanticipated Software Evolution，USE）。

我们将在本章给出一些针对 B3G 系统研究工作的综述。其中包括移动通信系统中软件演进的软件工程原理和技术，以及软件集成和私有化的动态适配。我们将自己的工作与面向方面的软件开发（Aspect Oriented Software Development，AOSD）以及 USE 等前沿研究进行了比较，并指出了高度分布化的移动电信系统开发如何从 AOSD 和 USE 部署中获益。

除了异构网络的无缝安全接入，我们在研究 B3G 系统时也考虑了服务的高可用性和提供给用户的最佳服务质量，这对系统提出了极高的需求。以下就是一些我们认为对 B3G 通信平台至关重要的核心议题：

1）快速开发和提供周期；

2）最小的系统崩溃时间；

3）实时更新和升级支持；

4）第三方组件和服务集成；

5）异构环境集成；

6）服务私有化；

7）上下文感知。

我们认为 DSA 无论在网络侧，还是终端侧对于我们提供解决上述问题的能力都意义重大。DSA 的目标在于推动服务和平台的演进，以支持不同部分以不同速度发展，以及便利私有化、上下文感知和泛在计算。我们认为生命期长、不间断运行且高可用性的系统将是从 DSA 获益的首选，这些系统可能是嵌入式的，也可能大规模广泛部署，而这些都是移动通信系统的特性。

现在部署的适配机制一般关注于内容，不太关注通信，并且几乎不关注服务逻辑以及行为本身。因此，内容和通信适配的研究远远超前于服务逻辑适配。简便起见，我们将使用服务适配这一术语表示服务逻辑或行为的适配。

与传统方式相反，我们将面向方面的编程（Aspect-Oriented Programming，AOP）[1-4]与计算反射、后绑定相结合来适配服务和服务平台，并做到不中断服务。我们尽可能将环境变化延后。本章剩余部分组织如下：9.2节列举我们提出的动态服务适配方法，重点解决模块化、变更点、AOP、后绑定和反射等问题；9.3节将给出我们研究平台的综述；在9.4节展示完应用于第三方服务适配和集成上下文的动态服务适配后，我们将做一下总结。

9.2　方法

首先明确我们关心的问题是服务适配的内容以及何时、如何完成适配（见图9-1）。服务适配的内容需要区分开软件系统计算、状态和通信的基本特性，何时完成适配主要解决软件开发过程中适配在系统中可操作的时间，而如何完成适配需要研究提高适配有效性的工具和技术。

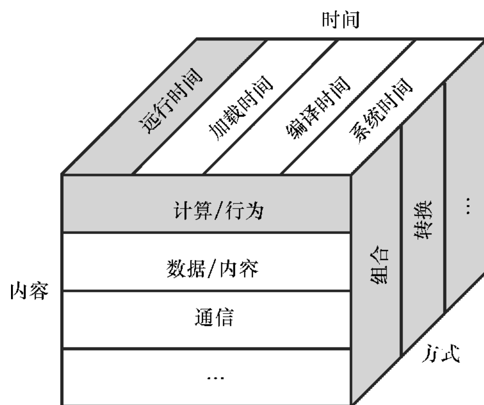


图9-1　适配维度

适配的概念是和模块化、变更点紧密相关的。模块化是一种改善系统灵活性和全面性，从而缩短开发周期的机制。变更点可以让我们在系统设计时显式指定模块边界，确定期望的变化点而无需显式定义这些变化。引入变更点的目的在于通过通用系统和变量系统方面的分离和组合使系统更加灵活。变更和变更点依赖于建立在系统之上的编程平台提供的模块化机制。大多数新建系统都是基于面向对象技术，系统中将类和实例作为模块化结构以及变化单元。AOP提供了一个更细致的新型模块化结构，这样我们可以表示单个实例的横切关注点。

大多数变化产生于系统初始部署之后，需要在系统生命周期的末段加以处理。一般情况下，我们倾向于通过实时按需完成大量纠正动作的方式避免系统发生停机现象。为了满足这一需求，我们认为反射架构和后绑定是DSA平台的两个关键点。

在我们提出的DSA方法中，我们采用方面模块化结构足以表示变化单元。

计算反射、动态 AOP 和后绑定使得我们在尽可能不造成系统停机和服务中断的情况下最迟地满足变化需求，完成服务和服务平台的适配。

下一节中，我们将简要介绍模块化、变更点、AOP、反射和后绑定等概念。

9.2.1　模块化和变更点

模块化是管理复杂性的一种方法。我们通过将一个复杂系统组织成较小的低复杂性子系统，然后依据特定原则重新组合子系统，来尝试改善系统的全面性和灵活性，进而缩短开发周期[5]。我们设计的模块对于每个复杂的设计方案或很可能变化的设计方案都是不可见的[5]。变更点（也称作热点[6]）使得我们在系统设计时可以指定模块边界，而这些边界就是我们期望发生但无需显式定义的变化处。我们借助变更点，就可以通过通用系统和变量系统方面的分离和组合在变化的上下文中获取灵活性。

例如，在面向对象编程中，模块化的基础单元就是对象。类对象用于捕捉实例的特性。实现特殊关注点的代码并不会在一个或少量模块中实现本地化，而是要传播至更多甚至所有地方，以横切其他模块，以及实现其他关注点。由于横切代码的非显式结构，它们难以用于推理，因此也难于变化。变化的一致性既难以证明，也难以执行。面向对象编程及其类模块结构虽然已被证实适用于很多建模场景，却无助于以充分模块化的方式实现日志方面的关注点。

变更和变更点依赖于建立在系统之上的编程平台提供的底层模块化机制。多数现代软件系统采用面向对象技术，将类和实例作为模块化结构（即变化单元）。尽管这样的粒度足以应对许多案例，但是实际应用中还是需要更细致的模块化方式（如方法实现）以支持更小语义单元的变化。同样，类和实例等传统模块可以支持初始系统的正常构造，而系统的后续变化可以横切这些模块，以影响更多位置。

9.2.2　面向方面的编程

AOP[1-4]是一种解决关注点分离（Separation Of Concern，SOC）[7]问题的新兴软件技术，其基本假设为横切是复杂系统的固有现象。AOP 解决上述问题的方法是引入新的（或另外的）模块化单元，以显式捕捉横切结构。这些结构被称为方面，可见于软件系统设计和实现中。

方面是代表横切关注点实现的模块化单元。它通过通知的使用将代码片段（出现连接点时需执行的代码）和连接点（代码执行时的精确执行点）关联在一起。由通知执行的一组相关连接点被称作切入点。连接点描述符用于标识编排过程的目标，而编排过程将计算变化应用于在通知对象中声明的底层系统。

将方面及其通知集成于底层系统的行为被称之为编排。一般情况下，编排可

实现于编译期、加载期或运行期。http：//eclipse. org/aspectj/就是一个编译期编排的实例。实例中，编排器解析了一个 AspectJ 程序，将 AspectJ 抽象语法树 AST 转换为有效的 Java AST[8]，然后生成用于标准 Java 虚拟机的 Java 字节码。Mangler（http：//javalab. cs. unibonn. de/research/jmangler/）[9]实现了 Java 类文件的加载期转换。而 AspectS（http：//www. hpi. uni-potsdam. de/swa/）[10]则采用一个运行期编排器依据相关方面对底层系统进行转化，其中被编排的代码基于方法封装[11]、反射[12]和元编程[13]实现。

当前已出现多种支持面向方面概念的方式，从通用的方面语言（如 AspectJ 或 AspectS）到领域特殊性的方面语言（如 RG[14]或 D[15]）都有相应内容提出。其中多种语言都支持对横切关注点的表示，并且可下至方法和实例变量层次的粒度。类似于面向对象编程中的对象，方面也出现于软件开发生命期的所有阶段。通常可观察到的方面实例包括结构性或设计约束、特征和系统特性或行为（比如差错恢复和日志）。

9.2.3 后绑定与反射

在软件开发和产品周期中我们经常发现一些我们希望在项目最初期就能了解明白的事情[16]。在系统设计时，经常会有一些需求无法理解透彻，而且在系统初始部署后，会发生很多在开始无法预测和解决的变化。相反，这种变化必须在很晚的后期解决，比如在部署后的产品期。如果在运行期能够实施绝大多数正确的操作，系统的停工期将大大缩短。

为了满足上述需要，我们认为反射架构和后绑定将是实现 DSA 平台的核心要素。

反射架构是系统通过加入表示自身（或方面）的结构而实现的[12]。这些结构的集合体被称作系统自表示，既可以帮助系统观察自身的执行情况和影响，也可以帮助系统改变自身行为。反射系统的前一种特性被称作自省，后一种被称作调解。在服务更新和适配的上下文环境中，自省使我们能够观察已部署的一组服务及其运行的计算环境的计算特性。然后调解就可以基于我们的观察结果，产生服务或系统的变更。由于还有一些研究是针对编译期反射（特别在产生式编程上下文环境中），如果在文中不显式指出，我们讨论的就是运行期反射。

后绑定描述了一种将决定推迟至后期时间点的机制。通过后绑定，我们能够避免过早做出用于设计决定的承诺，尤其是当决定与一些我们尚未确定是否维护的变更点相关的时候。前绑定需要我们在很靠前的时间点就提出解决可能变化的抽象，而后绑定则帮助我们避免做出不成熟的抽象。极端后绑定使得这些决定尽可能晚到运行期发生。

9.2.4　平台

通过 DSA 的应用，我们希望当由于系统变化需要实现适配时，服务和服务平台的适配尽可能晚地发生，这样可以避免系统的停工及其导致的服务中断。

为了推进我们的研究，需要理解完全动态系统的本质，以推进我们对待实现目标可能性以及难点的深入理解。我们所基于研究平台的选择和扩展是决定我们进步与否的重要因素。如图 9-2 所示，我们的平台组件依上下顺序构建，形成一个层次化的架构。

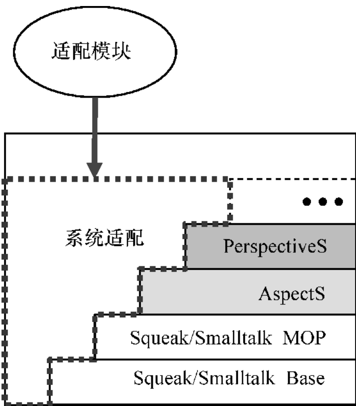


图 9-2　动态适配平台

通过在各种平台（见图 9-3，从服务器到小型设备）上运行 bit-identically，Squeak/Smalltalk 为我们提供了一个非常动态的面向对象的多媒体脚本环境[18]。其最突出的一些特点包括覆盖自省和调解的广泛反射支持、强大的元对象协议（可为我们提



图 9-3　运行于手持设备的 DSA 平台（原图来自参考文献［13］，已获得 Journal of communications and Networks 的授权）

供对自身平台计算特性的完全访问）及其对深度后绑定的支持。深度后绑定可将绑定决定推迟至真正需要的时间点。元对象协议的思路是语言可被打开，直到用户可以调整设计和实现，从而使得语言或环境满足用户的特殊需求。通过上述过程，可以鼓励用户积极参与语言的设计过程。基于元对象协议的语言设计可采用面向对象系统的方式实现，这样就可以利用面向对象的优点，使得语言设计的特性更灵活。

AspectS 对 Squeak/Smalltalk 环境进行了扩展，以适应实验性的面向方面系统开发。AspectS 的目标是在动态系统的上下文环境中，提供一个用于发掘面向对象软件组合的平台。它可以支持简化的指导性元层次编程，用于通过提供与方面相关的模块来解决混乱代码现象。由于 AspectS 不依赖于代码转换（不论源代码还是字节码），体现出很好的灵活性，但实际上它采用了元对象组合。与其他大多数注重类层次方面的 AOP 方式不同，AspectS 支持实例层次的方面，这样就可以实现横切一组独立实例的行为的模块化。

PerspectiveS 建立在 AspectS 的基础之上，可支持在 Squeak 环境下的动态行为分层。它可以协调一组方面的上下文感知，从而使得我们可以使用依赖上下文的行为优化系统，且不需要底层系统开发者关注可能的优化。PerspectiveS 使得底层系统的关注点更大地与其依赖上下文的行为产生分离[19]。这样，底层系统就可以免于提供相应行为，以显式地针对在开发或部署期都未知的上下文变化。PerspectiveS 利用保持对象身份的多角色动态组合，推动了角色建模。角色可以按需增加或删除，同时每个角色都引入了自己的一组状态和行为。

所有上述层次都允许我们既可以实现我们的基本服务逻辑，也可以在需要的情况下，将服务逻辑适配于另外的需求和不可预知的环境。基于我们研究平台的动态本质，适配行为可按需动态地开展，而我们的服务已经实现部署和激活。

在下一节，我们将使用第三方服务集成的场景来展示我们提出的 DSA 平台的应用。

9.3 案例：第三方服务集成

我们希望 B3G 移动通信系统是向第三方服务提供商开放的，从而可以使他们提供自己的服务。然而，并不是所有提供的服务都会精确符合运转的服务平台，因此，为了最终给用户提供愉快的服务体验，必须做一些适当的调整。虽然可预先确定和应用一些调整，但许多调整的需求在初始服务部署之后才出现，这也许不会造成现行服务的中断。

接下来，我们将通过讨论在第三方服务集成场景中发生的 4 个具体情形，来

展示 DSA 的价值。我们选取的情形如下：

1）辅助安全措施；

2）设计指南的一致性；

3）后用户界面的品牌化；

4）升级、更新和补丁。

上述列表还很不完整。可考虑增加的主要候选部分包括系统扩展、使用说明、计量、个性化、预标准发布以及管理要求的满足等。

所有在第三方服务集成、安全措施介绍、类型指导的一致性、UI 品牌化、升级、更新、补丁等上下文环境中有关 DSA 的认识都来自与第一手经验，并通过原型实现获取。在此，我们将 Squeak、AspectS 和 PerspectiveS 都作为我们的实验平台。Squeak 使用后绑定和反射便利提供了一个反射性在线环境。AspectS 在动态系统中增加了量化和遗忘，而 PerspectiveS 则提供了上下文感知的匹配激活机制。

9.3.1　基础服务

我们决定提供一个称作个人数字助手（Person Digital Assistant，PDA）的新服务，这一服务是用户希望在其移动终端上使用的。我们称在服务平台上采用的特殊 PDA 实现为“福雷”（Fauré）（http：//russell-allen.com/squeak/faure），它是一个运行于手持设备的开源 PDA 实现。

获取实现该 PDA 服务的第三方组件实现很简单。我们首先在组件仓库（即 Web）中定位组件，然后下载并安装到我们的服务平台中，这样就可以将其作为我们的集成测试床。运行中的“福雷”PDA 将在其欢迎屏幕汇总待做的项目和某一天某一时刻预定的事件。

这样，我们就可以立刻开始使用我们的新 PDA，组织待办事件的列表、个人时刻表、联系信息或者接下来的社会活动。我们的新 PDA 同样可以提供一个用于勾画便签或小图样的小型记事本、用于欣赏音乐的小钢琴、突出我们的这个新东西功能的 3D 演示以及只是为了玩游戏的可选组件。

9.3.2　辅助安全措施

直到现在，我们在将 PDA 组件集成到我们服务提供环境的过程中尚未遇到问题。然而，当我们点击退出键终止 PDA 服务时，却显示出这个第三方组件原始开发者设定的一个假设在我们的平台上是不适用的。这就是退出 PDA 时不仅仅会终止 PDA 服务，还会终止 PDA 的执行平台，以至我们的整个服务平台。

为了保证上述现象不在产品环境中发生，我们需要调整退出键功能块的行为。我们没有要求“福雷“（Fauré）开发者修改相应组件以满足我们的需求，

也没有亲自动手修改组件源码，我们决定通过使用 DSA，以一种非扩散的方式（即其不会影响原始实现中的源代码）实现适配。我们提供了一个适配组件（即一个方面）去协同原始组件，并指导我们的实时环境在退出键功能中插入另外的行为，这样一来，每次用户需要终止 PDA 服务时，都会被提示他们希望只终止 PDA 服务，还是同时终止整个运行平台（见图 9-4）。

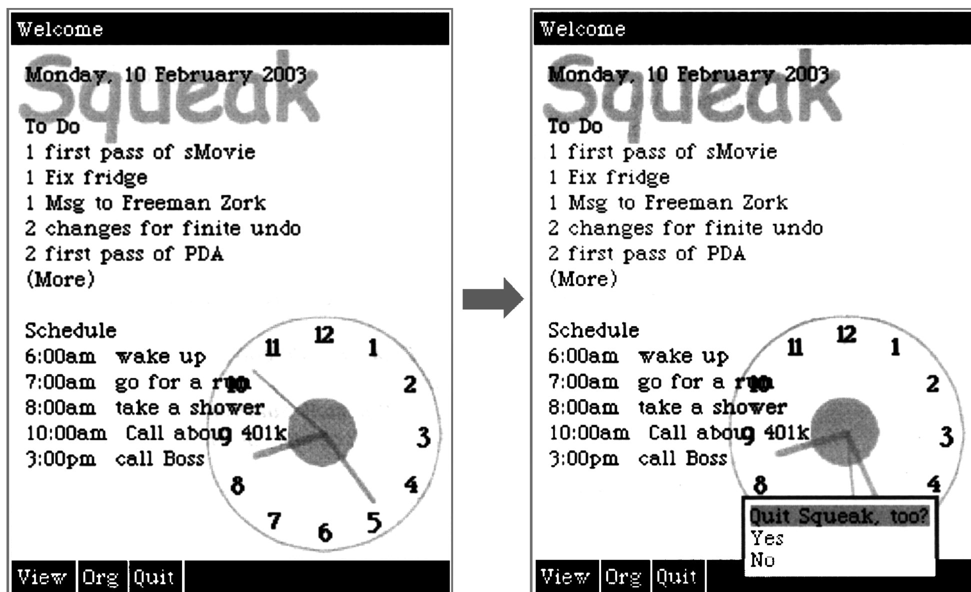


图 9-4　辅助安全措施

在本例中，我们引入了一个辅助对话框，以更直观地表现应用的变化。在商业系统中，我们不太可能提供这样一个选项，但在终止 PDA 服务时，还是需要给出一个不终止我们的平台（此处为 Squeak）的选项。

图 9-5 和 9-6 中的列表展示出适配过程是如何实现的。图 9-5 表示当客户点击退出键时会调用的方法：即 PDA 会存储当前状态，然后，调用我们平台提供的退出原语（即 Smalltalk quitPrimitive），随之终止整个平台（即 Squeak）。

```
FaureWorld class>>
quit

    PDA current saveDatabase: 'db.pda'.
    Smalltalk quitPrimitive.
```

图 9-5　退出原语的调用

图 9-6 展示了用于实现前面所述方法的适配模块部分（FdsaQuitAspect）。通

过使用 AspectS（http://www. hpi. uni-postdam. de/swa/）[10]，我们没有调用Faure World 的原始退出方法，而是构建了一个提供代码执行的通知（AsAroundAdvice）。类似于前面的实现，在存储 PDA 状态后，我们插入了一个对话框（self confirm:），用于询问客户是要只终止 PDA，还是终止整个平台（Squeak）。

我们在部署 PDA 服务时加上述适配模块，用于指导我们的服务平台执行所期望的步骤。

```
FdsaQuitAspect>>
adviceBrowserBuildMorphicSystemCatList

    ↑ AsAroundAdvice
        qualifier: (AsAdviceQualifier
                attributes: { #receiverClassSpecific. })
        pointcut: [OrderedCollection
                with: (AsJoinPointDescriptor
                        targetClass: FaureWorld class
                        targetSelector: #quit)]
        aroundBlock: [:receiver :args :aspect :client :clientMethod |
                | ctx morph |
                PDA current save Database: 'db.pda'.
                (self confirm: 'Quit Squeak, too?')
                        ifTrue: [Smalltalk quitPrimitive]
                        ifFalse: [self deleteFaureWorld]]
```

图 9-6　用于退出的安全措施对话框

9.3.3　设计指南的一致性

许多运营商要求第三方服务通过其基础结构提供服务时，要符合特定的用户界面设计指南。突出的例子包括 NTT DoCoMo 的 i-mode 设计指南和 Vodafone 的 Vodafone live 设计指南。与设计指南不一致会导致用户的误解，并最终让人感觉服务提供商并没有适当提供服务包。这将会影响用户的接受度，小到个人服务，大到整个服务包。设计指南相关的适配不但对于原来没有依据特定设计指南开发的第三方组件是必要的，对于现存设计指南或策略发生变化的情况也是如此。

在我们的例子中，我们选取了一种设计指南，即出现在退出键上的文字需要用红色着色。Fauré PDA 的开发者并没有考虑从用户角度出发更改退出键文字的颜色，因此他们也没有提供相应的修改方法。实际上，退出键的着色隐藏在 PDA 组件的 UI 初始化序列中。

图 9-7 展示出我们做过适配修改后的 Fauré UI。我们采用了一种非扩散的适配模块，将退出键上文字的颜色修改为红色。

从图 9-9 中，我们可以看到适配模块的部分内容（FdsaQuitButtonMigrateAspect）。

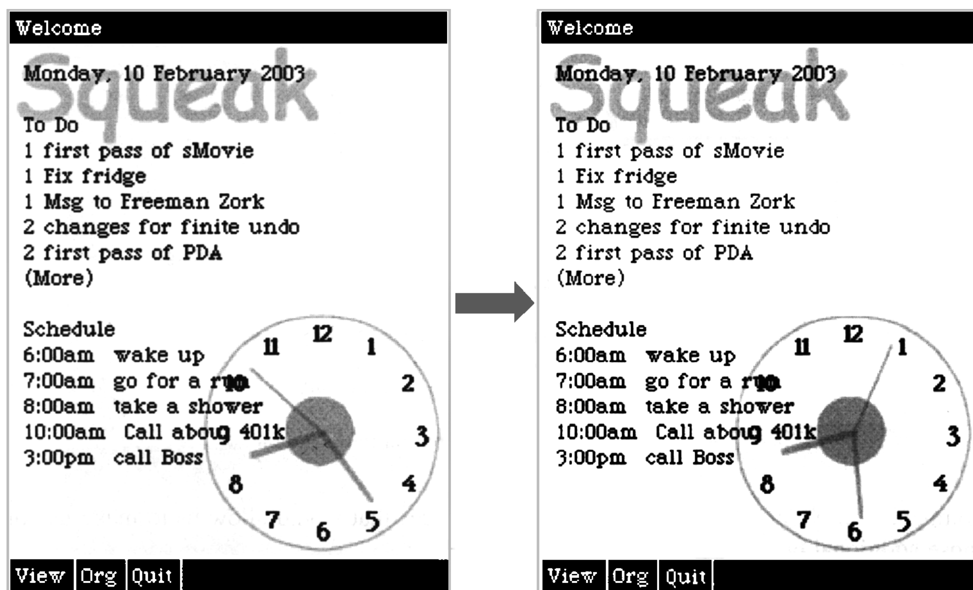


图 9-7　强制的设计指南

此部分内容用于指导 Fauré 菜单条按钮的构建方法（见图 9-8），即我们在每次调用 FaureMenuBar 的 addButton：withAction：target：方法后，都会生成提供执行代码的 AspectS 通知（AsBeforeAfterAdvice）。我们的代码将验证实际构建的按钮是否为退出键，如果是就将其文本颜色更改为红色（m color：Color red）。

```
FaureMenuBar>>
addButton: aName withAction: aSymbol target: aTarget

    | m |
    (m ← SimpleButtonMorph new) label: aName;
            borderWidth: 0;
            target: aTarget;
            actionSelector: aSymbol;
            actWhen: #buttonDown;
            cornerStyle: #square;
            color: Color black;
            height: 20;
            vResizing: #rigid;
            hResizing: #rigid;
            layoutInset: 3;
            changeTableLayout.
    (m findA: StringMorph)
            color: Color white.
    self addMorph: m.
```

图 9-8　按钮初始化

虽然这种变化只会在 PDA 组件的启动期有效，设计指南相关的调整也需要应用于所有运行中的 PDA 组件，这也就包括那些已经启动并且已经开始运行其 UI 初始化序列的 PDA 组件。为了实现上述目的，我们从运行平台的反射特性受益匪浅。我们采用了一个元程序，去发现所有不符合我们设计指南要求的运行 PDA，然后转换所需的所有地方，使得所有现存退出键的文本用红色着色。

```
FdsaQuitButtonMigrateAspect>>
adviceFaureMenuBarAddButtonWithActionTarget

    ↑ AsBeforeAfterAdvice
        qualifier: (AsAdviceQualifier
                attributes: { #receiverClassSpecific. })
        pointcut: [OrderedCollection
                with: (AsJoinPointDescriptor
                        targetClass: FaureMenuBar
                        targetSelector: #addButton:withAction:target:)]
        afterBlock: [:receiver :args :aspect :client :return |
                | m |
                m ← receiver submorphs first findA: StringMorph.
                (m notNil and: [m contents = 'Quit'])
                        ifTrue: [m color: Color red]]
```

图 9-9　特殊化的退出键启动

9.3.4　后用户界面的品牌化

许多第三方组件提供了可用于辅助品牌化的 UI 元素。服务平台运营商或服务提供者可以利用这些元素放置品牌名称、商标甚至广告。可惜大多数时候，组件提供商并没有提供显式接口，以使我们能够利用那些品牌化的额外机会。

DSA 可用于增强基础 UI 的功能，它可在 UI 窗口小部件以及其他表面部分绘制辅助的品牌化相关信息，而无需显式地提供接口以完成上述任务。

Fauré PDA 提供了一个 3D 演示，去展现其使用的 Squeak 环境所具备的高性能 3D 绘制能力。这个演示展示出一个六面绘成不同颜色的立方体，滑轮控制可以放大缩小和沿任意方向旋转该立方体。

由于立方体的表面部分采用了普通的纹理，它可以作为辅助品牌化的首选。我们提供了一个适配模块，在其表面放置另外的纹理（即 DoCoMo 欧洲实验室的标识）（见图 9-10）。我们提出的适配应用突出特点在于动态性和非扩散性，因为它可在运行期执行和调用，并且它不需要为了适应我们的当前需求更改原始组件的源代码。

图 9-11 展示出用于初始化 Fauré 3D 演示场景的代码。在这里生成了一个 3D 场景对象（即一个立方体），并将其加入实际场景，同时不向立方体的任何面绘制特殊纹理。

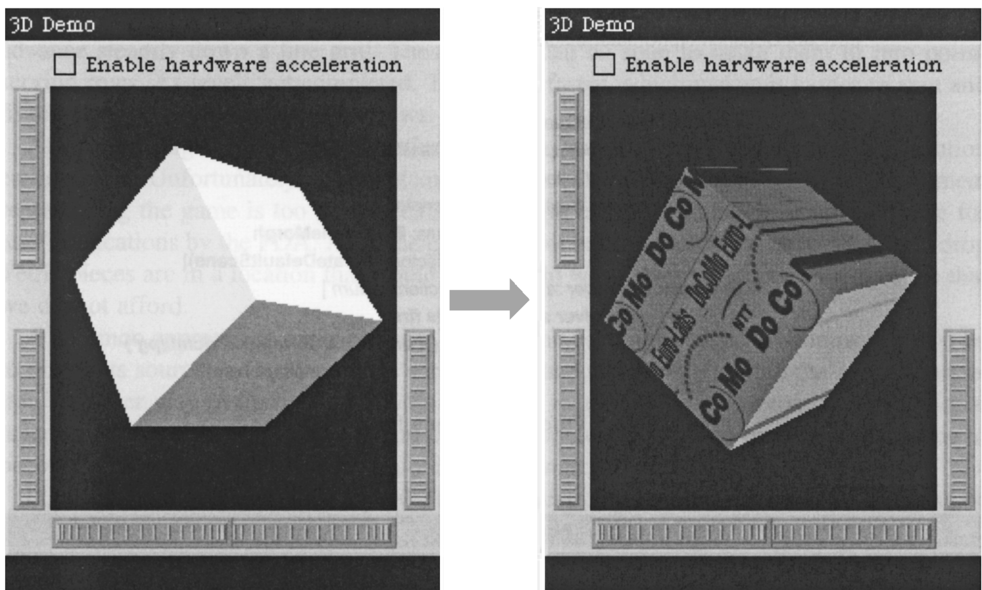


图9-10 后用户界面的品牌化

```
B3DSceneMorph>>
createDefaultScene

    | sceneObj camera |
    sceneObj ← B3DSceneObject named: 'Sample Cube'.
    sceneObj geometry: (B3DBox
            from: -0.7@-0.7@-0.7 to: 0.7@0.7@0.7).
    camera ← B3DCamera new.
    camera position: 0@0@-1.5.
    self extent: 100@100.
    scene ← B3DScene new.
    scene defaultCamera: camera.
    scene objects add: sceneObj.
```

图9-11 3D场景样本的生成

在图9-12中我们采用了另一个适配模块（FdsaDcml3dMigrateAspect）生成相应的通知（AsBeforeAfterAdvice），该通知用于在利用（createDefaultScene）生成3D演示场景后加入待执行的代码。然后，我们将DoCoMo欧洲实验室标识作为新纹理提供给3D演示对象。

这个特殊适配也是满足实例或状态迁移要求的实例。这个要求对于调整具有状态的现存对象是必须具备的。这些对象拥有的状态实际是在我们的适配模块激活前发生的副作用的结果。

```
FdsaDcml3dMigrateAspect>>
adviceB3DSceneMorphCreateDefaultScene

    ↑ AsBeforeAfterAdvice
        qualifier: (AsAdviceQualifier
            attributes: { #receiverClassSpecific. })
        pointcut: [OrderedCollection
            with: (AsJoinPointDescriptor
                targetClass: B3DSceneMorph
                targetSelector: #createDefaultScene)]
        afterBlock: [:receiver :args :aspect :client :return |
            receiver scene objects first
                texture: ((Form fromFileNamed: 'dcml.jpg')
                    asTexture wrap: true)]
```

图 9-12　为 3D 立方体提供纹理

9.3.5　升级、更新和补丁

通过观看图 9-10 中的 DoCoMo 欧洲实验室标识，我们发现了一个 3D 着色错误。这个错误并非来自 Fauré，而是在我们的实时环境中就已经存在了。既然已经发现了这个问题，那么我们就可以立刻很好地修复它，而不需要重建整个系统，停止需要修复的所有节点，使用新建系统替换旧系统，然后再次启动。注意系统关闭和再次启动可能需要我们备份和重置操作状态。

我们没有采用重建并更换系统的过程，而是提供了一个动态适配模块，用于在系统运行时完成对 3D 着色问题的修复（见图 9-13）。

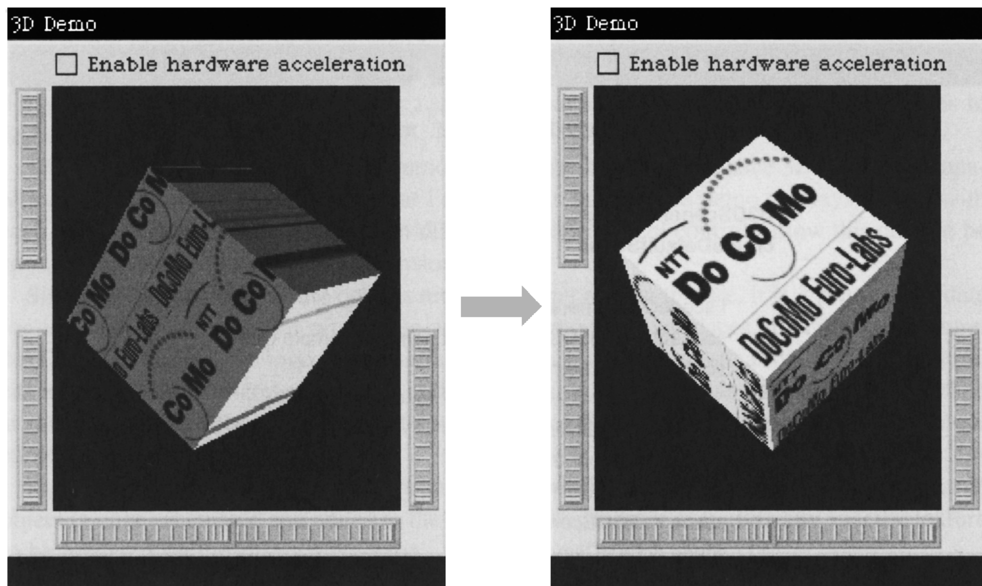


图 9-13　修复 3D 着色问题

9.3.6 服务集成

Fauré PDA 还提供了一个称为 Same Game 的小游戏，这个游戏原来是 Eiji Fukumoto 为 UNIX 和 X 平台编写的。Same Game 的任务是通过减少面板上的砖以获取更高的积分。砖是可以选择的，并且可以通过点击邻近几个具有相同特征的砖来进行消除。但是如果我们的大多数客户想去玩其他更流行的游戏（比如说俄罗斯方块）怎么办？俄罗斯方块最早是由 Alexey Pazhitnov 和 Vadim Gerasimov 在 Electronica 60 上开发的。俄罗斯方块游戏中，在屏幕上方会出现具有规则形状的积木块，并有规则地向下落入格中，可以将积木块旋转，使其组成得分的横线。当过完一关后，俄罗斯方块就会加速，使得旋转和堆积成线的难度加大。

在搜索俄罗斯方块的具体实现后，我们发现一个可用于我们执行环境的应用。遗憾的是，这个实现并不适用于我们的 PDA：由于表示游戏的 UI 元素高度超出了 PDA 中用户应用提供的高度，UI 元素显得过大。此外，用于旋转和扔落俄罗斯方块的游戏控制按钮放置的位置会导致我们浪费难以负担的更多的屏幕空间。

要使得俄罗斯方块这个新游戏适于 PDA 环境，通常做法是获取其源代码，修改代码，然后重新生成游戏应用。另一种使得俄罗斯方块符合我们需求的做法是提供一个辅助部分的软件，以指导我们的运行环境如何对游戏进行转换，使其可在我们的提供环境中部署。

图 9-14 中既展示了上面所述的俄罗斯方块的原始应用，也展示了经过转换

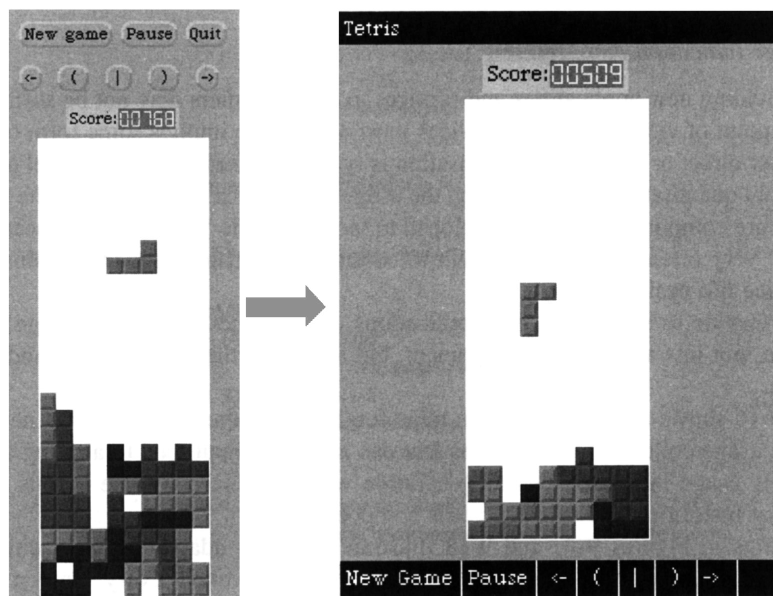


图 9-14 俄罗斯方块适配和集成

并集成到 PDA 中的同一个应用。图中可看出如何对应用的尺寸进行变化，以满足 PDA 的限制条件。此外，所有原来放在游戏区上端的游戏控制按钮，现在已放置于 PDA UI 的按钮行。如果这个游戏在一开始就是为 PDA 设计的，那么现在放置的位置也应是首选位置。

使俄罗斯方块适用于 PDA 环境，并不仅仅是证明其集成是可行的，还需要使用户便于使用。为此，我们不得不对我们 PDA 的起始菜单进行扩展，提供一个运行俄罗斯方块的可选入口。图 9-15 展示了扩展后的菜单，我们增添的俄罗斯方块新入口位于列表的最末段。

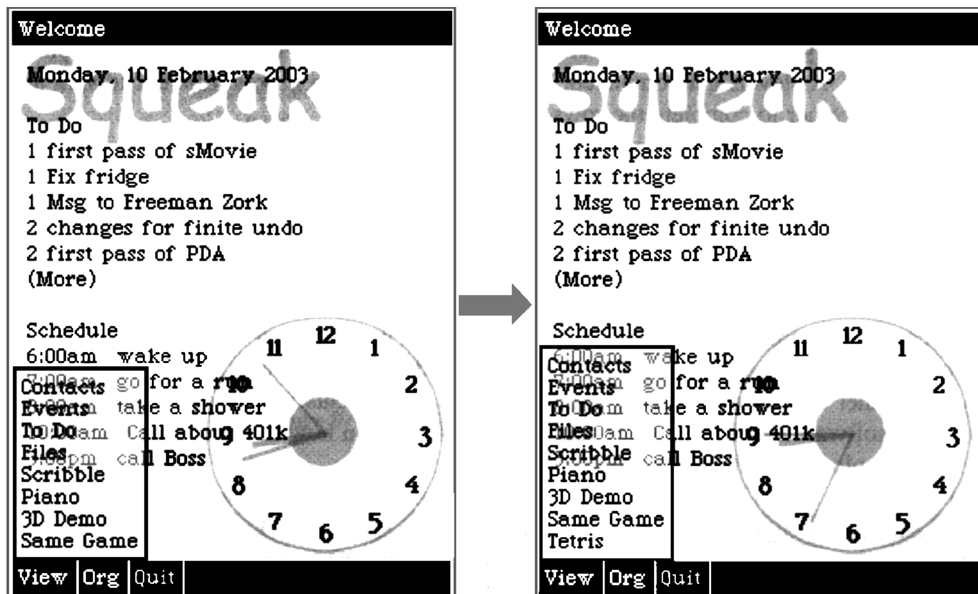


图 9-15　增添俄罗斯方块菜单入口的 Fauré 界面菜单

9.3.7　使用说明和计量

从商业角度考虑，仅仅为我们的用户提供新应用和服务是远远不够的。大多数情况下，提供服务意味着某种形式的补偿，可以是直接的，也可以是间接的。补偿通常建立在服务水平协议（Service Level Agreement，SLA）的基础之上，SLA 会描述服务使用时的相关定量信息。由于大多数时候第三方软件组件并不会针对特定的 SLA 进行开发，同时由于 SLA 也会经常变化，在服务生命期中过早提出针对特殊使用说明的承诺是毫无益处的。

DSA 使得我们不但可以在应用和服务开发完成后指导它们去提供使用说明信息，也可以在它们部署后，甚至晚至运行期，完成相同的工作。

图9-16展示了一个俄罗斯方块的使用说明跟踪报告。跟踪过程中，每次新游戏的启动都会报告给使用收集机制，这个机制可以作为输入向分级和计费引擎发送信息。这个使用说明记录生成功能是通过一个用于指导俄罗斯方块原始组件的适配模块引入的。

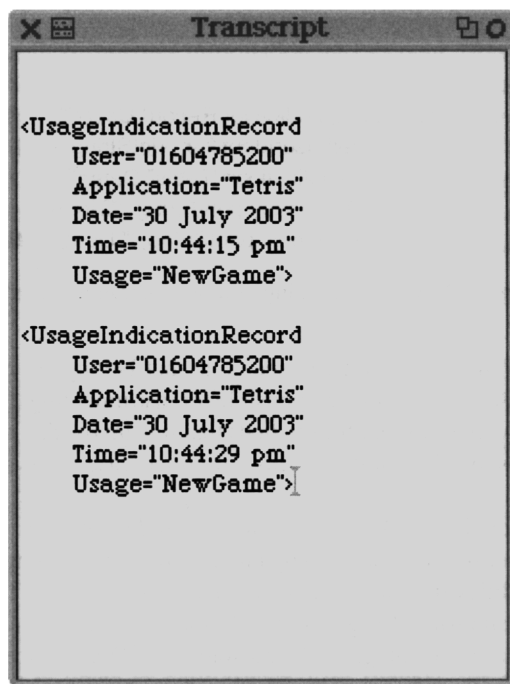


图9-16　输出的俄罗斯方块使用说明记录

图9-17和图9-18中的列表展示出适配过程是如何实现的。第一个列表（见图9-17）表示每次客户点击“New Game”按钮（TetrisBoard >> newGame）时调用的方法。此时，俄罗斯方块就会开始一个新游戏。

```
TetrisBoard>>newGame

    self removeAllMorphs.
    gameOver ← paused ← false.
    delay ← 500.
    currentBlock ← nil.
    selfscore: 0.
```

图9-17　开始一个俄罗斯方块新游戏

在下一个列表（见图9-18）中，可以看到属于我们的适配模块（FdsaTetrisUsageAspect）并且负责指导新游戏方法（new Game Method）的代码。指导过程的实现方式是每次（除了第一次）方法被调用时，都会向相应的负责实体（最简单的情况下，应为系统记录及Smalltalk控制台）发送一个使用说明记录。

```
FdsaTetrisUsageAspect>>adviceTetrisBoardNewGame

    ↑ AsBeforeAfterAdvice
        qualifier: (AsAdviceQualifier
                attributes: {#receiverClassSpecific. })
        pointcut: [OrderedCollection
                with: (AsJoinPointDescriptor
                        targetClass: TetrisBoard
                        targetSelector: #newGame)]
        afterBlock: [:rcvr :args :aspect :client :return |
                thisContext baseSender baseSender selector
                        ~~ #initialize "the first game is for free"
                                ifTrue: [self postTetrisUsage]]
```

图 9-18　输出俄罗斯方块使用说明记录

图 9-19 展示出实现的一个简单方法 postTetrisUsage。

```
FdsaTetrisUsageAspect>>postTetrisUsage

    Transcript
        cr; show: '<UsageIndicationRecord User="',
                self userIdentifier printString,
                '"Application="Tetris" Date="',
                Date today printString, '"Time="',
                Time now printString, '" Usage="NewGame">'.
```

图 9-19　使用说明记录的打印输出

我们部署的 PDA 服务同时包括用于集成的俄罗斯方块组件和需完成相应功能的适配模块。上面展示的适配模块只负责动态使用说明记录的生成，而本章中并没有讨论将俄罗斯方块集成进 PDA 服务所需的适配模块。

9.4　展望——面向上下文的编程

在目前以信息为中心的环境中，上下文信息发挥着越来越重要的作用。上下文感知应用和服务即将成为不同服务提供商的区分器。从具有环境依赖性的基于位置的服务，到高度个性化的服务，越来越多的新服务的出现为运营商在市场竞争中提供了主要的竞争优势。当前市场充满了对固定费率基础设施的期望。

目前为止，不少开发上下文感知软件的尝试都采用了相当耗时并易于出错的过程，最终生成的解决方案也限制重重，过于专用化。这种情况下，上述解决方案既需要很长的市场推广周期，又经常由于专用性，难以为未来的上下文感知服务提供借鉴。

我们相信，通过确定上下文感知服务及其显式定义以及将其引入我们的软件

平台基础架构的底层原理，可以为服务提供商开发上下文感知服务和服务运营商提供相应服务有效地降低门槛。如果在开发时间和部署时间上可以得到充足的支持，我们就可预见，门槛的降低会带来更多的上下文增强的服务，使得服务包向更有意义的方向扩展，从而扩大用户群，提高用户满意度，并提高用户粘性。

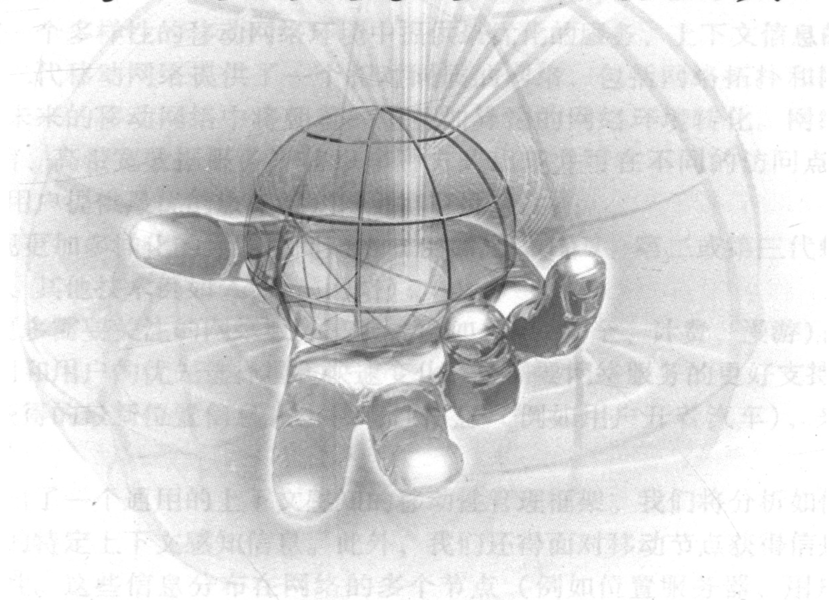
我们把实现上述目标的方式称为面向上下文的编程或COP[20]。我们和学术伙伴将要建立一个面向上下文编程的思想，并使之超越传统的编程模式。当前已有几个计算机科学的传统研究领域涉及到适应性和适合性概念，这些概念中已考虑到上下文敏感等内容。然而我们的方式并不拘泥于特定应用的解决方案，而是要开发一个新的编程模式，并支持改善软件可维护性、健壮性和重用性的编程语言结构和软件基础架构，而这些特性都是需要适于高度动态环境的。这样我们就可以使得服务提供商和运营商免于不合适的技术困扰，以一种简练方便的方式表达多种多样的应用和服务的面向上下文特性。

9.5 小结

我们相信下一代移动通信系统将会比以前更加复杂。这不但由于系统所连接环境的复杂性日益提高，也部分归功于这种系统的相关假设，即系统对第三方服务提供商开放，用于使用某个运营商的通信平台向终端客户提供相应的服务。随着服务提供商的变化，服务包不断依据客户的需要和偏好进行调整，这样服务水平协议将成为区别服务的核心因素之一。由于变化是个普遍现象，并非特殊情况，并且变化很难预期，我们需要采用新概念、新机制和新技术来支持向更好的方向变化，以实现所有的目标。我们需要探索，要想使变化令人满意，到底需要什么。我们还必须明白如何推进我们的计算平台满足我们的需要，以便我们能够将其移植到一个不同但更好的平台。要是这种平台不存在，我们还需要明白如何去构建一个这样的平台。除了概念、机制和技术，我们还需要合适的底层架构支持传播适配模块、调整其激活和停用状态、检测并解决冲突（如果需要的话），解决与移动代码相关的安全性和保密性问题。通常情况下，变换激活和停用状态或者适应性组合是超越于针对基于语义服务组合的基本途径的。我们利用COP的支持，就可以使用一种简捷的途径来表示各种服务和应用的面向上下文特征。我们认为我们的研究工作会提供一种更富有规律性的DSA实现途径。

第3部分

环境中的服务和智能嵌入



第 10 章　上下文感知的移动性管理

Christian Prehofer

对于将来的移动系统来说，上下文感知是一个重要的增强部分。其目的在于使移动通信系统和用户使用环境相适配。这一章将重点阐述上下文感知的移动性管理。

为了在一个多样性的移动网络环境中提供最优化的服务，上下文信息的使用是关键。第二代移动网络提供了一个相对同质的网络，包括网络拓扑和网络服务。然而在未来的移动网络中将朝着一个更加异构的网络环境转化。网络服务（多媒体通信、高带宽数据服务）将以多种方式出现并可在不同的访问点获得。这样如何向用户提供最优的服务将变得更富有挑战性：

1）出现更加多样化的无线访问网络（例如无线 LAN、第二或第三代蜂窝网络及其变种、其他技术例如 Ad Hoc 网络）；

2）有更多需要关注的网络服务事项（例如 Qos、安全、计费、漫游）；

3）应用和用户的优先选择项将快速变化，这需要网络服务的更好支持；

4）用获得的最新位置信息和组移动性信息（例如用户开着汽车），来支持服务。

我们提出了一个通用的上下文感知的移动性管理框架。我们将分析如何获得与切换相关的特定上下文感知信息。此外，我们还得面对移动节点获得信息的复杂性和混杂性。这些信息分布在网络的多个节点（例如位置服务器、用户配置服务器、接入点），甚至是移动节点，包括应用层。它们需要在特定的时间提供给特定的节点。此外，上下文信息通常比网络层功能发展地更迅速。为了应对上下文信息的变化，我们需要可扩展的软件技术，包括软件代理、活动网络等。

通过利用移动设备或移动网络所蕴含的上下文信息，可有效改进移动设备的性能。这些具备上下文感知能力的网络设备，其行为能自适应地与所在环境的上下文信息进行适配。这些信息包括用户、移动设备和移动网络的上下文信息。

上下文信息可以从大量的不同类型的来源获得，例如用户配置管理系统、位置管理系统、流量监测和感应系统。在移动网络中所面临的挑战首先是从多样化的来源中收集信息并进行处理，然后将处理后的信息发布到多种不同类型的客户端，这些客户端可能运行于差异很大的移动或固定设备之上。

围绕上下文感知所进行的大量工作，其目的主要是为各种应用提供支撑。因

此我们将通过移动性管理和区域分页等途径来实现，集中精力于如何加强移动网络的基本网络层功能。我们将讨论在网络基础设施层面有哪些特定的问题需要解决。网络层同应用层在几个方面存在差别：首先是网络功能持续运行时间更长，且针对特定的网络进行了优化；其次，网络功能对于实时类型的操作，如交互式通信系统，进行了高度优化。从这个角度出发，上下文信息管理需要具备很高的性能。最后，移动网络需要面对脆弱的，有时候甚至是缺乏的无线资源，将来的移动环境将包括各种异构的网络和服务。

在10.2节我们将呈现具备上下文感知的网络实例，以及讨论移动网络中上下文管理所面临的问题。在10.3节将分析在这些实例中进行上下文管理的不同选项，我们将展现一个灵活的能够动态安装软件模块的方法，即代理，它能够很好地满足我们的需求。随后将在10.4节引入一个通用的上下文管理框架，它提供几种最优化网络服务上下文管理的机制。特别地，我们还区分了原始的、应用独立的、应用特定的上下文处理方法。在10.5节，我们描述了一个可扩展的体系架构，它实现了由Wei等人首次引入的移动网络上下文管理框架。为了能够使应用相关的服务配置具备灵活的扩展性，该体系架构将包括一个服务配置基础设施，它能够在可编程的网络节点上安装和配置可定制的模块。考虑到网络元素的性能，一个可编程的平台能够以有效的方式实现网络层特定模块。

10.1 上下文感知的移动性管理实例

在这一章中，我们将介绍一些关于上下文感知的网络功能的实例。这将有助于阐述我们上下文管理功能的主要需求。

10.1.1 上下文感知切换

接下来我们将介绍上下文感知切换的例子。通过此例，说明借助于用户的上下文信息，切换可以得到优化。切换的目标是找到一个最优的新接入点（AP），如图10-1所示。

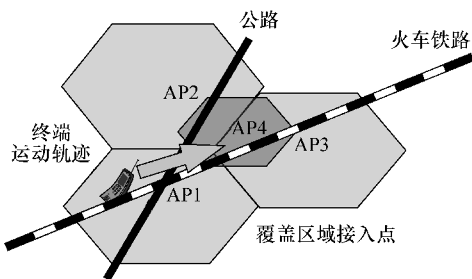


图10-1 上下文感知的切换
（来自Prehofer等人[2]）

通常，用于切换决策的信息是信号的强弱和无线资源的可用性。然而，仅靠这些信息无法进行有效的切换。因为即使一个AP在本地测量中略好于其他点，但基于此所做的切换决策未必就是最好的。

例如在图10-1的场景中，一个

节点移动到 AP2 和 AP3 都能覆盖到的区域。如果具备正确的位置预测或其他上下文信息，那么就能做出正确的切换选择。如果节点在火车上，切换到 AP3 或 AP4 是更好的选择，哪怕是 AP2 目前具有更强的信号。

在许多场合中考虑节点移动和用户偏爱选项，切换会得到最优化。例如，一个节点在汽车或火车上，它的路径就被限制在特定的区域内。节点本身会包含一些信息显示其是否在汽车或火车内，从节点的移动模式就可以大致分析出其在乘坐哪种类型的交通工具。

快速决定是否进行切换也很重要。然而节点配置和位置信息常常保存在核心网络的中央服务器上，对于判断是否进行切换，远程检索这些信息就会显得太慢了。另外切换时所在位置的无线信号可能是比较弱的，也会对检索产生影响。

在将来的网络中，另一个重要问题是搜索采用不同技术的接入点，其代价是沉重的（耗费大量计算资源以及能量)，对于只支持一种模式的设备来说甚至都无法进行搜索。在网络的上下文感知功能的帮助下，避免不必要的 AP 搜索就显得很有用了。在了解到有可能的 AP 信息后，重新配置到别的模式的概率就大大降低了。

解决的方案就是准备好有助于切换决策的上下文信息。一个典型例子是当前的移动模式（例如获得其是在火车上还是公路上)。实现中可以基于需要的上下文信息在节点上配置不同的算法。这一实现方法需要一个跨层接口以搜集不同层的信息，然后部署一个决策算法作出最优决策。

除了将用户用于连接 SIP 网络的终端或者软件定义为用户代理，SIP 网络还定义了如下的逻辑实体：代理服务器、重定向服务器和注册服务器。

10.1.2　定制寻呼服务

当具备寻呼系统后，移动节点的位置注册更新只需在移动到不同的寻呼区域时才执行，如图 10-2 所示。选择正确的寻呼区域的大小和形状是确保寻呼系统效率的核心。一方面，过大的寻呼区域增加了寻呼过程的代价；另一方面，过小的寻呼区域会增加重新注册的概率以及电池能量的消耗。当前使用固定大小的寻呼系统在许多场合下并不是最优的。接下来提出的上下文管理体系架构实现了寻呼区域的定制化，能改进寻呼的性能。

在移动网络中，网络需要知道移动节点的位置，以便于和其保持连接，这需要移动节点不停地向网络报告其位置。一个快速移动的节点不得不频繁地进行位置更新，会导致相当大的信令负荷。基于此，使用寻呼技术替代这种机制则可以节省能量，降低位置更新引起的信令负荷。

采用寻呼技术时当节点移动到一个新的寻呼区域后，需要向网络发起一个比

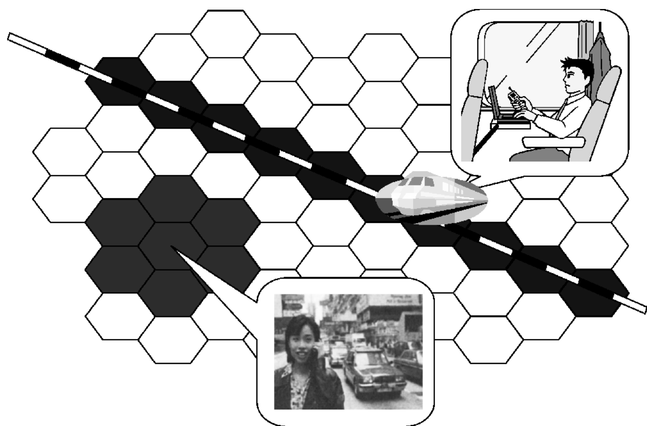


图 10-2 上下文感知的寻呼机制（来源于 Prehofer 等人[3]）

较复杂的位置注册过程。一个寻呼区域包含几个蜂窝，也包含多个 AP。为了接收电话，寻呼过程需要找到节点在寻呼区域中的确切位置。寻呼的策略可以是“Blanket Polling”或“Sequential Paging”。在前一个策略中，寻呼的请求将同时发送给寻呼区域的所有无线 AP，由这些 AP 回复寻呼请求。在后一个策略中，寻呼请求是按照节点位于某个蜂窝的可能性大小按顺序发送给 AP 的。

如上所述，寻呼区域的最优大小和形状是寻呼系统效率的核心。主动网络技术将有助于寻呼区域的定制。类似于定制化切换，定制化寻呼利用用户配置信息和移动信息来动态调整寻呼区域的大小。

图 10-2 给出了一个定制的区域寻呼例子。假定节点在火车上，同时它移动的方向和速度是确定和已知的。在这个例子中，没有必要在铁路线外的区域进行寻呼。最优的寻呼区域是如图 10-2 所示沿着铁路线进行划分，因为火车移动速度较快，因此采用一个大的区域寻呼标准是比较有利的，可避免频繁地位置更新。否则对于移动缓慢且不可预测（例如一个行人的移动电话）的节点，其最优的区域寻呼标准应当是以该节点为中心，而且是几个较小的寻呼区域（因为注册的代价比较低）。从上述的例子可以看出根据节点的移动信息，使用动态自适应的寻呼区域将更加有效。

已经有一些文献在讨论寻呼区域的优化。基于接收的采样样本可以记录一些参数，如速度和方向等，然后根据这些参数计算寻呼区域。在 Wu 等人[6]的文献中提出了一个基于行为的策略方法，通过收集节点长期的移动日志来估算节点的位置。在 Lei 等人[7]的文献中提出了一个类似的方法，根据不同的用户配置信息，输入参数可以进行动态调整，显然，在网络层和应用层之间进行交互是必须的。

10.2　移动网络中的上下文管理

这一部分将讨论在搜集、预处理、分发和使用上下文信息来增强网络服务中所遇到的问题。通常，上下文信息可以是静态的也可以是动态的，可以来源于不同的网络位置、不同的协议层和设备实体。

为优化网络性能，切换到一个新的接入点或接入网络，需要根据正确的时间和地点，准确使用上下文信息。最优接入点的选择也取决于用户上下文信息，例如用户设备或开销限制。接下来我们将讨论与切换决策相关的典型的上下文信息，以及向节点提供这些信息会遇到的困难。通常上下文信息来自不同的地方、协议层以及设备实体，例如第一层和第二层为物理和链路信息，应用层信息则可来源于移动设备，也可来源于网络端。

表 10-1 给出了一个典型的上下文信息的分类。这个表只是某个时刻的信息，有可能出现新的上下文信息类型，例如在以后的 Ad Hoc 网络，可能出现如用户组之类的上下文信息。一些项如用户配置可能出现多次，因为这些信息可能存在于用户设备、运营商和一些服务提供商。

表 10-1　上下文信息分类

	移动设备的上下文信息	网络侧的上下文信息
静态	用户环境配置信息	用户配置信息和历史信息
	应用配置信息	网络定位、网络能力和服务
静态	可访问的接入点	潜在的下一个接入点
动态	应用的请求	位置信息和定位预测
	设备状态（电池、接口状态等）	网络状态和网络负荷

作为一个例子，这里的用户配置包括定制的服务和服务优先选择参数（如当资源不够时哪些服务可以降级或中止）。

问题在于这些上下文信息通常是有关联的，使用起来也比较困难，原因如下：

1）对于许多网络层任务来说，时间都是很有限的，做决策需要非常迅速。此外，也许在做决定的时刻网络信号没有或很弱。为此，上下文信息的搜集和提供需要及时且快速，且在无信号或信号弱之前进行。

2）上下文信息是分布化的，在单一的网络实体中并不能完全获得，例如，一些上下文信息存在于用户的家庭网络中，一些可能存在于漫游的网络中，或者一些就在节点内部维护着。

3）动态的上下文信息变化频繁，随着时间的推移准确性会降低，例如，获

得当前AP的负荷信息是比较有吸引力的，但其有效性也随着时间的推移迅速降低。

4）上下文信息本身以及解析上下文信息的方法随着时间也一直在变化。为此需要新的上下文数据解析方法。例如考虑以下的情况：

① 用户配置信息的变化（例如新的服务出现，需要选择新的接入点）；

② 关于服务变更的漫游协议；

③ 位置预测方法的演进。

上下文切换主要的思想在于使用上下文信息来避免随意和错误地切换，否则将降低用户体验。当然不是所有情况下信息都是明确的，从而能确保选择是最优的，信息传送给设备以及信息的处理也会带来额外的负担。

在大多数情况下，应用只使用特定设备的小部分上下文信息。在移动网络中还存在一些特定的问题，例如网络不稳定、无线链路资源不足、电源限制以及漫游通信。此外，不同类型的设备和通信系统也需要考虑在内。

10.2.1 上下文管理和上下文感知切换的相关工作

Stemm 和 Katz[8] 以及 Pahlavan 等人[9] 已经考虑了更加智能化的切换过程。然而用于协助切换决策的参数仍然仅限于无线接入技术相关的信息，如信号强弱。

Stemm 和 Katz 对于不同类型的网络使用了不同的切换策略，考虑了包括空中接口类型以及接入路由器的带宽等因素。Chan 等人[10] 使用了模糊逻辑来处理从各种不同无线接入网络接收的复杂信息，核心在于切换使用的算法以及如何进行模糊逻辑的应用。另一个相关的文献是 Kounavis 等人[11] 所提出的一个可编程切换的框架，包括移动控制、网络控制、网络和移动辅助的切换。我们的工作可以对此进行扩展以适应上下文信息的变换。总结一下，这些基本观点是对于各种不同的接入网络需要更多智能切换策略，但是它们都没有给出一个能适应各种环境的切换机制通用框架。

10.3 上下文感知切换的上下文管理方法

在这一部分将展示不同的途径来实现上下文感知的切换，也将看到各种各样的上下文信息是如何被网络和移动设备处理的。这个例子也将有助于我们理解后面一个更加通用的框架的主要例子。我们认为，主动地收集用户上下文信息是为切换做准备的，网络中的上下文信息可能会同时涉及到多个实体，例如位置信息服务器。本节内容是由 Prehofer 等人[12] 所出版本的一个扩展。

在我们的实现方法中，会使用到以下几个主要实体，如图10-3所示。

1）切换决策点：它将决定是否切换到某个 AP，例如图 10-3 所示的移动节点。

2）网络侧的上下文信息收集点：它收集和编辑来自不同地方的相关信息，并可能传送给切换决策点。

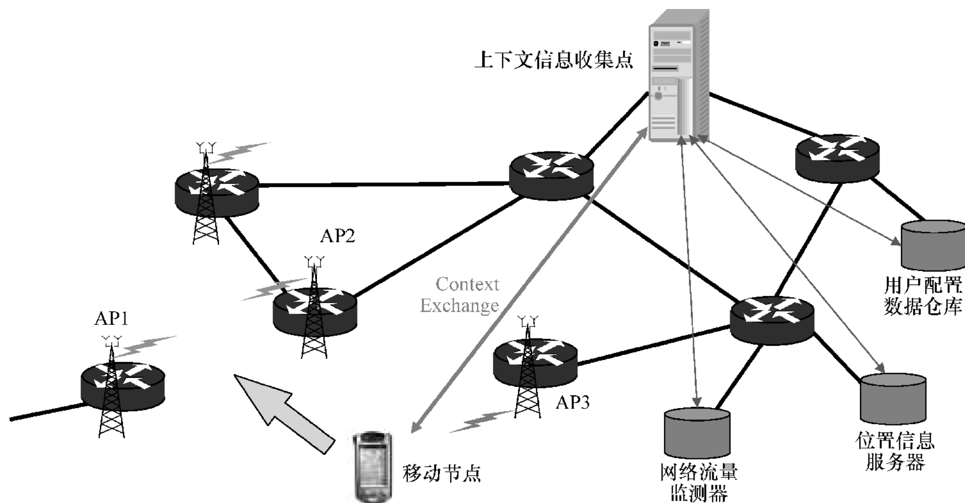


图 10-3　上下文感知切换的体系架构

我们假定切换决策发生在移动模式，但由提供上下文信息和规则的网络进行控制。这很重要，因为许多重要的动态信息只能在节点获得（例如信号强度），网络侧控制的级别取决于移动节点逻辑实现以及它如何应用网络上下文信息。

接下来我们所面对的问题就是实时和有效地提供切换决策点。另外如何解析上下文数据的逻辑需要预先确定好。我们需要在几个有冲突的需求中寻求平衡。一方面，提供尽可能多的相关信息或在有变化的时候更新信息。为了适应新的需求（处理新型的上下文信息），节点的算法和逻辑实现应当是灵活和通用的。另一方面，系统应该具有可扩展性，同时把负荷降至最低，包括通信开销和计算资源。

接下来我们提出两种切换决策的方法，并将它们整合到一个可扩展的框架内。

10.3.1　上下文交互协议方法

传统的上下文信息交互方式通过定义相应的格式和属性来交互信息（例如使用 XML CC/PP[13]）。此外，还需要在上下文信息收集点和移动节点间传送上下文信息的协议。这意味着上下文信息的解析应当固定，可能的话还可对其进行

标准化。相应的算法需要在节点内实现。而在网络侧收集上下文信息的算法可以针对网络或位置具体部署。

总而言之，需要考虑下述因素并预先部署在节点中：

1）网络侧和节点协商好上下文格式和属性。

2）节点的上下文解析和切换决策的算法。

3）上下文收集机制和如何编辑以网络侧指定格式表达的信息。

4）交互上下文信息的协议。

除了算法外，何时触发上下文信息的收集和传输也是很重要的，它有时是由上下文驱动的，例如一个节点离开一个蜂窝小区时将触发对于其他上下文信息的收集。

进行上下文切换时，下述步骤需要由切换决策和收集点执行：

1）在切换决策时刻前交互上下文数据。

如果需要可重复发送上下文数据的更新消息。

2）当连接断开或需要不同的接入点时，移动设备执行下述切换过程。

① 收集移动设备的动态上下文信息（例如可查看到的接入点、信号强弱、应用状态）。

② 依据收集的数据和上下文信息启动决策算法来选择合适的切换接入点。

③ 切换到选择好的接入点。

举个例子，参考图10-1所示的场景，假定上下文格式就是一个关于AP及其QoS能力的优先选择表。上下文信息在网络端进行编辑，优先选择表能够对基于用户或网络上下文信息的用户设备进行优化。AP的能力见表10-2。假定用户在行驶的汽车内，参数是语音。上下文信息收集点首先收集和编辑如下的信息：

1）基于移动方向的预测，比较好的AP是AP2和AP4。

2）基于服务参数对其进行排序：AP2、AP4。

切换前列表（AP2，AP4）将发送给移动设备。切换过程中，移动设备的算法将根据可使用AP的动态信息，从中选择参数最高的一个AP。在这个例子中，控制功能将在网络侧实施，发送给节点的数据量很小。

表10-2 接入点能力描述表例子

	空中接口类型	运营商	QoS1（音频）	QoS（视频）	QoS（数据）
AP1	UMTS	A	+	+	+
AP2	UMTS	B	+	+	+
AP3	GSM	B	+	−	−
AP4	WLAN	A	−	+	+ +

由移动节点承担更多功能的算法变化如下，假定传送的上下文数据包括表格中所列的属性，那么决策算法可以参考更多的动态上下文信息（应用请求、会话的 QoS 级别）。在这种算法中，上下文信息的收集点需要做更多的预测和 AP 的排序，并把表格中 AP 的上下文数据发送给移动设备。

在两种实现方式中，发送给节点的信息都是针对设备进行了优化的，这样能降低每个设备传输和处理的代价。对于一些涉及到多个节点的上下文信息，可以通过广播的方式发送给附着在一个接入点的多个设备。其他不涉及到多个节点的信息，可单独发送给特定的节点。

两种实现方式的局限性在于节点的上下文信息处理是固定不变的。当有新的上下文信息可利用时，网络需要预先把这些信息转换成预定的格式和属性。由于一个移动设备的生存期限为几年，因此这点的确是一个缺陷。该缺陷将在后面的部分进行处理。

10.3.2　上下文感知决策代理的动态下载

在接下来的实现方式中，我们不以明确的格式将上下文信息发送给移动节点，而是通过下载一个代理软件，它包含了上下文信息以及解析这些数据的逻辑算法。通过这种方式，我们避免了固定上下文信息交互协议所带来的局限性。在可扩展性方面，这种方式优于前面的方法，它不需要上下文信息格式及其解析逻辑的标准化。对于这种方式我们假定切换决策点是可编程的，以支持代理的动态安装。

方法包括下述步骤：

1）在上下文信息收集点准备一个具备相应算法的代理，并且在切换决策点收集需要的上下文数据。这些是在上下文环境明显发生改变时启动的，例如进入一个新的蜂窝或者用户的配置信息发生变化。

2）切换决策点切换前下载代理。

3）切换时激活代理：

① 输入：动态的节点上下文信息。

② 输出：切换决策（例如选择的接入点）。

相应的流程可参见序列图 10-4。在第一步中代理的准备工作可以由多种方式实现，一个选择项是根据环境从已有的库中进行选择，此外软件可以进行参数的动态配置或根据特定的规则从工具库中进行组装。这种方式的主要优点在于可扩展性。我们假定只有一个软件代理可以安装，例如是一个移动 Java 平台。这个代理必须具备访问上下文信息和控制切换的接口。此时代理的逻辑接口成为一个主要的限制因素。

回到切换的例子，假定用户正在一辆行驶中的小汽车上，其优选的服务为低

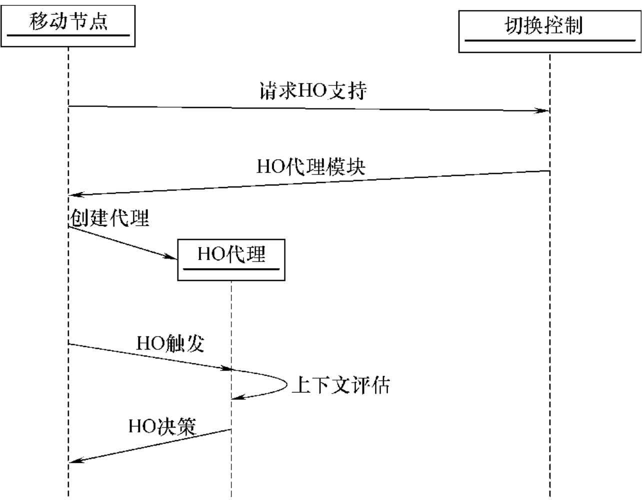


图 10-4　基于代理的上下文感知切换序列图

耗的语音服务和偶尔的快速数据服务。在这种情况下，只有当选择数据服务时，切换集合点决定选择 AP4。从网络下载的代理的决策算法如下：

```
Procedure CalculatePreferredAP
Dynamic Local Input：RequestedUserService
If (RequestedUserService = Data)
  then preferredAP = AP4
  else preferredAP = AP3
```

注意到代理的激活点是在切换的时候，与之前的实现方式作比较，只有代理需要下载，而不是一个完整的表。代理可以看成是一个简单的脚本，它包含了一个根据核心的上下文信息所编辑得到的决策树。它取决于实际运行的环境和语言，这样一个代理也只比采用列表描述的上下文信息大一点。

其他更多的演进算法如下所示：

```
Procedure CalculatePreferredAP
Dynamic Local Input：RequestedUserService,RequestedDataRate
If (RequestedUserService = Data)
  then if RequestedDataRate < 50Kbit/s
     then preferredAP = AP3
     else preferredAP = AP4
else
If (RequestedUserService = Vedio AND RequestedDataRate > 15Kbit/s)
     then preferredAP = AP2
     else preferredAP = AP3
```

这里传送的代理大小只比之前的例子大一点，同样也包含决策的逻辑信息。通常，代理在印象中都比较大，并且能容纳更多更明确的上下文数据和算法，这也是代理的一个优势。其包含信息的复杂性和数量可根据用户和用户环境进行调整。与之前类似，也可以考虑以广播的方式将代理在同一时间发送给多个节点。

一个尚未考虑的问题是使用代理的解决方案的安全性问题，我们需要避免错误的或恶意的代码。假定代理的执行是在一个安全的软件运行环境，进一步来说，代理的结束运行也是可以保证的。对于这一方案，一个潜在的缺陷是下载的代码带来的负担。在许多情况下，传输数据的总量等于或小于第一种方案。这就需要下载合适的代码，而且我们可以利用一些标准的组件，它们不必每次都下载的。

10.3.3　综合方案

以前的方案都是比较极端的方案：第一个方案中上下文信息需要在网络和所有移动设备之间达成一致的静态格式和规范；而第二个方案中为了解析和处理上下文信息，移动设备需要从网络中下载代理。区别点在于代理方案对于上下文信息的数据结构和格式来说是可扩展的。相对而言，第一个方案限制于固定的数据结构（例如一个表或一个决策树）。

接下来我们将把这两种方案结合起来，在不同的时间段将算法和上下文信息下载下来。我们使用一种能够进行上下文信息交互的代理，同时只在上下文信息发生结构变化或处理逻辑发生重要变化时才更新代理。

完整的处理过程包括下述两个步骤，可在不同的时间段单独执行。

1. 用于交换代理的步骤

1）在上下文信息集合点准备一个代理，包括算法和在切换决策点收集的上下文数据。

2）切换决策点下载新的代理，包括算法和上下文信息交互协议。

2. 上下文信息交互和切换的步骤

1）如果上下文环境发生改变则更新上下文数据，例如进入了一个新的蜂窝单元。

2）启动代理执行切换决策过程：

① 输入：动态的节点上下文信息。

② 输出：选择的接入点。

上述过程如图 10-5 所示。

我们重复使用以前的例子，假定代理如下述的伪代码所示，使用了 4 个参数 APL1、APL2、APL3 和 APL4，它们是接入点的一个列表，并根据 AP 的邻近关

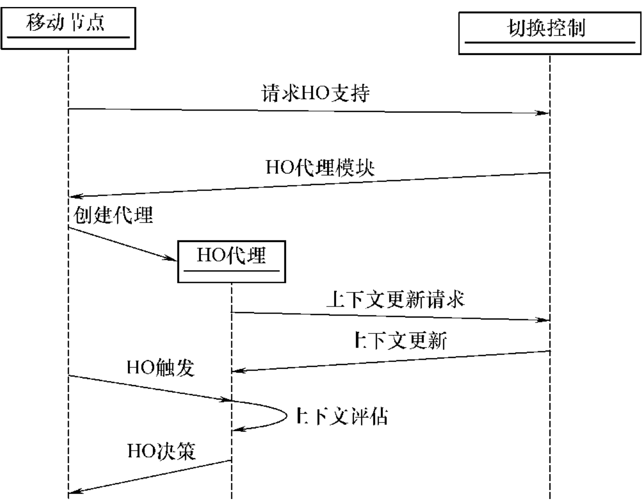


图 10-5　基于代理的上下文交互序列图

系和能力进行一定的排序。主要的流程是计算切换优选的 AP 列表，如果列表是空的，那么可任意选择一个 AP 进行接入。

```
Process ExchangeContext
  Repeat
  Wait_event from ContextCollectionPoint
  Receive APL1,APL2,APL3,APL4 from ContextCollectionPoint

Procedure CalculatePreferredAPList
Dynamic Local Input: RequestedUserService,RequestedDataRate
If (RequestedUserService = Data)
  then if RequestedDataRate < 50Kbit/s
     then preferredAPlist = APL1
     else preferredAPlist = APL2
 else
 If (RequestedUserService = Vedio AND RequestedDataRate > 15Kbit/s)
     then preferredAPlist = APL3
       else preferredAPlist = APL4
```

其中代理的更新可以在不同的时间段进行，例如在每次用户上下文环境发生改变时，比方说用户进入一辆小汽车。当然也可以在用户或网络服务发生变化时更新代理，这种情况通常一年才出现几次。

10.4　移动网络上下文管理的体系架构

基于上一节对于上下文感知切换的讨论，我们的目标在于设计一个适合不同

网络功能的通用上下文管理体系架构。该体系架构的第一个版本由 Mendes 等人[15]提出。在这个架构中，能够收集、处理和分发非结构化的以及有效期短的信息，这些信息可存在不同的网络节点和设备。这个架构也应该能确保不同的应用能在合适的时间获得合适的信息。设计这样一个体系架构面临的一个主要问题是需要考虑不同的应用需求，这些需求可能涉及到信息交互模式、上下文信息的数据模型或者传输数据的大小等，其中最后一项对于无线网络来说尤为重要。稀缺的无线信道资源限制了移动设备交互信息的数量。另外一个重要需求是使用跨层接口在不同协议层有效收集信息的能力。

接下来我们将引入一个能够管理动态分布式上下文环境的通用架构，它不仅能在应用层，而且也能在网络层和链路层进行管理，它将应用层特定的上下文和通用的上下文区分开来。特定的和通用的上下文区分可以实现一个通用机制来收集上下文信息。通用上下文信息的调整由特定的适配器实现，以满足应用的具体需求。

我们这里提出的架构其主要目标是便于大量应用能收集、预处理、分发和使用不同的上下文信息。构建一个通用架构的第一个障碍在于上下文信息本质上是和应用相关的。为解决这个问题，通用架构分为两个部分，一部分模块是针对所有类型的应用，另一部分是针对特定应用的，如图 10-6 所示。

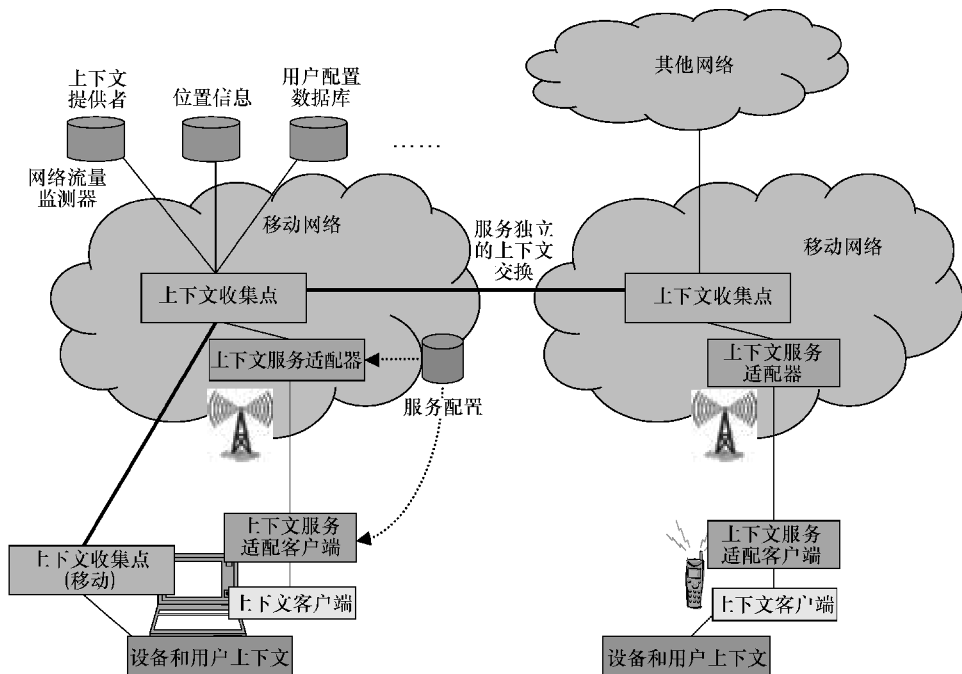


图 10-6　上下文管理架构的组件（来自 Mendes[15]）

在本节中我们将侧重于应用特定的适配机制，而不详细介绍独立于应用的上下文交互。它要求有一个上下文信息的通用模型以及有效的可扩展的信息处理方法。

由于每种应用都有它自己的特征，原始的上下文信息需要转换成应用可以理解的格式，还需要以每个应用都支持的协议进行交互。为此，在我们的通用架构中，上下文信息的收集由GCCP（Generic Context Collection Point）执行。一个GCCP能够和不同的上下文信息提供者以及其他GCCP建立连接，如图10-6所示。针对特定应用的原始信息调整由CSA（Context Service Adapter）实施。

收集的原始信息分布在不同应用中。每种应用在架构中被称为CC（Context Client），都有它自身的特定需求。例如不同的应用可以要求不同的数据模型（如层次性或扁平型）或者有不同的传输模式（如客户-服务器模式，点对点模式）。

由于GCCP负责提供原始的上下文信息，因此可能有一系列不同的CSA和同一个GCCP进行交互。每个CSA发送适配后的信息给负责和CC交互的CSAC（Context Service Adapter Client）。在架构中GCCP和CC之间存在两种组件，这使得上下文信息的分发和利用变得更加灵活。

由于不同类型的CSA和CSAC模块可以在不同时间和地点使用，因此在上述管理架构中包含有一个SDF（Service Deployment Framework），它负责控制网络中特定应用服务的提供。

10.4.1 移动网络中管理上下文信息的体系架构

这一节我们将阐述在上下文管理架构中，用于实现在移动网络中配置和管理不同类型服务的体系架构。在这一架构中，应用特定的适配器将依据需要由服务配置架构进行安装。图10-7显示了在拥有两个移动网络和两种服务类型环境中的架构。

GCCP间上下文信息的交互是独立于应用的，然而交互协议的配置取决于GCCP所在的位置。例如位于不同运营商网络的GCCP间通信很可能是受限的，而和移动设备中的GCCP进行通信则可能比较顺畅。

为了应对无线链路和设备稀缺的资源，当CSCA位于移动设备时，所提出的框架将把大多数应用相关的计算放在网络侧的CSA进行。

移动设备所需的上下文信息不但可以由本地的GCCP发起请求，或者由移动设备所登录的网络侧GCCP发起请求，还可以由其他网络的GCCP发起。从其他网络进行检索上下文信息将取决于不同运营商的策略。例如不同运营商也许会同意交换网络能力相关的信息，但对于用户的参数和配置信息则往往不会用于交互。

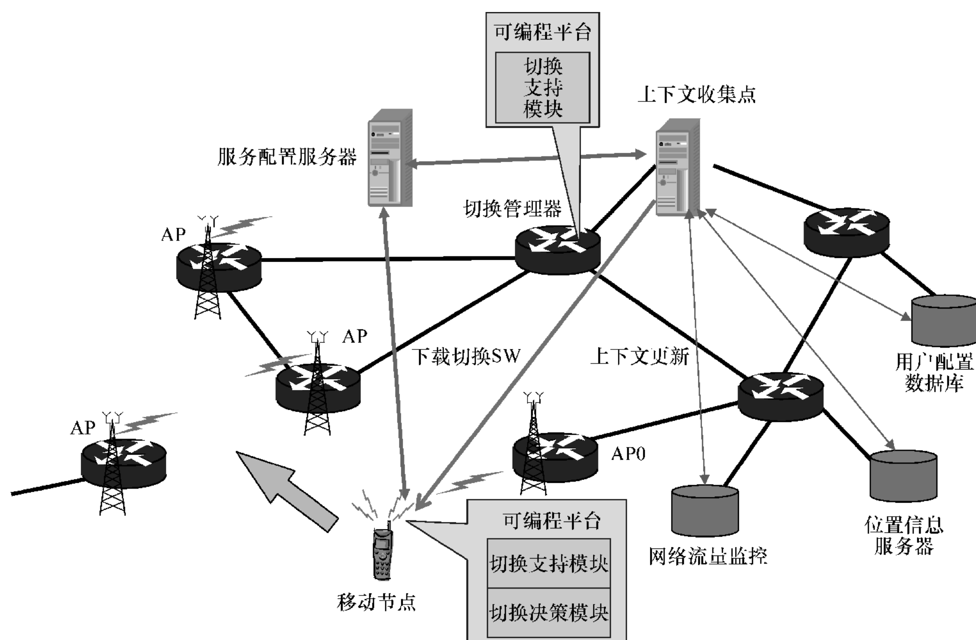


图 10-7　移动网络中管理上下文信息的网络架构

CSA 和 CSAC 和应用相关，它们可能还和特定运营商相关。这意味着那些漫游到不同网络的移动设备也许不一定能够使用在原有网络中应用的 CSAC。另外在外部网络的 CSAC 和本地网络的 CSA 之间的通信也有可能受限。

然而这是假定网络提供商只能将 CSA 配置在它自己的网络，以及将 CSAC 配置在所需的移动设备。SDF 的实现要求在网络中有一个 SDS（Service Deployment Server），如图 10-7 所示，它将存储运营商网络中所有可用服务的描述以及控制这些服务的配置。

10.5　上下文感知移动性管理的实现

对于实现上述架构，我们将重点集中在如何整合不同组件。在真实环境中，需要处理以下几个问题：首先软件模块的配置是可管理、可扩展的；其次我们需要为移动节点选择一个运行环境；再次，移动节点的配置和运行需要能够有效利用资源，并且考虑设备性能的配置；最后，处理流程需要是无缝整合，不能中断服务。

接下来我们将把上下文管理平台、一个活动平台和服务配置方案整合，以提供上下文切换所需的功能。上下文切换方案负责搜集并管理与不同服务相关的上

下文信息；活动平台用于交换和处理上下文信息；服务配置方案用于同步和管理相关活动节点的工作。

在如图10-7所示的网络架构中，上下文信息存储于LIS（Location Information Server，位置信息服务器）、NTM（Network Traffic Monitor，网络流量监视器）和用户配置等数据库中。LIS负责网络中每个移动节点的定位，以及掌握相邻AP的情况，而用户配置数据库则存储用户配置信息。NTM用于监测不同AP的可用带宽。此外我们引入了一个HM（Handover Manager，切换管理器），它负责控制一些接入网络的切换，负责过滤和处理HO相关的上下文信息。最后，SDS（Service Deployment Server，服务配置服务器）用于管理和安装网络节点和移动节点需要的服务模块。

移动节点包含两个模块：HSM（Handover Support Module，切换支撑模块）以及HDM（Handover Decision Module，切换决策模块），这是客户端服务适配器。网络端的切换管理器包括一个HSM，这是上下文服务适配器。

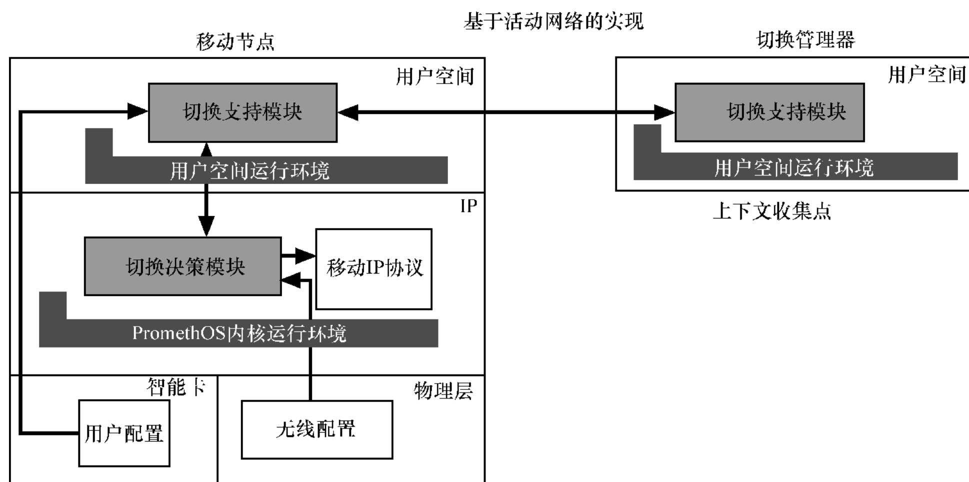


图10-8　上下文感知切换的可编程软件模块

图10-8展现了节点的运行平台，它也是活动节点的运行环境。例如，为了实现一个上下文感知的HO服务，我们需要在相关的节点，如移动网络和HM的活动平台上，安装一个上下文交换协议模块HSM（Context Exchange Protocol）以及一个切换决策模块HDM（Handover Decision Mechanism），这由服务配置功能实现。

10.5.1　动态节点平台和服务配置

接下来，我们将描述体系架构中的节点平台。通过使用动态网络技术来满足

之前提出的需求。决定切换到哪个最优的 AP 所执行的算法与上下文相关，因此涉及的网络元素和移动终端系统需要是可编程的。动态网络技术是一个比较理想的候选技术。我们的活动节点包括基本的处理硬件、节点操作系统和应用运行算法的运行环境。节点需要支持切换决策模块在运行中的动态安装，而不中断节点的工作。我们选择的动态节点体系架构和实现是 PromethOS[17]，它是一个基础平台，其在 Linux 环境下运行应用，允许用户空间或内核空间模块的按需安装。

PromethOS 提供了一个基本的动态节点管理平台，我们的应用场景需要支持服务组件的选择、安装、配置和管理。随后的章节将讨论这些问题。

10.5.2　节点服务配置

节点的服务配置包括服务组件的选择、下载、安装和配置实现，例如这些组件如何共同提供服务。Chameleon 服务配置框架很适合于完成这些节点服务配置的功能，它使用了一个由网络层服务配置所产生的服务规范和一个活动节点内在属性描述表，它用于决定活动节点上安装的服务组件实现。服务规范是一个 XML 文档，Chameleon 通过服务规范来解决节点属性的描述，创建了一个表示服务所有可能实现方式的树状结构。我们在应用 Chameleon 时，这样的服务规范由网络层服务配置方案产生，并支持上下文感知的切换。

10.5.3　上下文感知切换的网络层服务配置方案

网络层服务配置负责从网络侧向节点提供软件服务。服务配置分为两种：提供者驱动和用户驱动。对于前者类型，在任何用户到来前，服务预先配置在网络侧；对于后者来说，是根据需求来配置的，第一个用户到来时将启动网络层服务的安装。

我们使用了一个简单、中心化的网络层服务配置方案，它的核心是一个中央管理实体，称为服务配置服务器（Service Deployment Server，SDS），如图 10-7 所示。SDS 包括一个服务配置管理模块，它将控制网络范围内的信令和所有服务配置过程中的同步功能。SDS 也包含一个存有网络所能提供服务的描述符的服务器和一个代码服务器（包含服务组件的实现）。这些服务器由 SDM 统一管理，同时可以安放在网络任何地方。

10.5.4　整体流程

上下文感知切换测试床的完整操作流程如图 10-9 所示。第一步是服务配置，包括获取正确的服务组件，然后安装在合适的网络节点并进行配置。第二步是收集相关的上下文信息。第三步是上下文信息评估和切换决策。

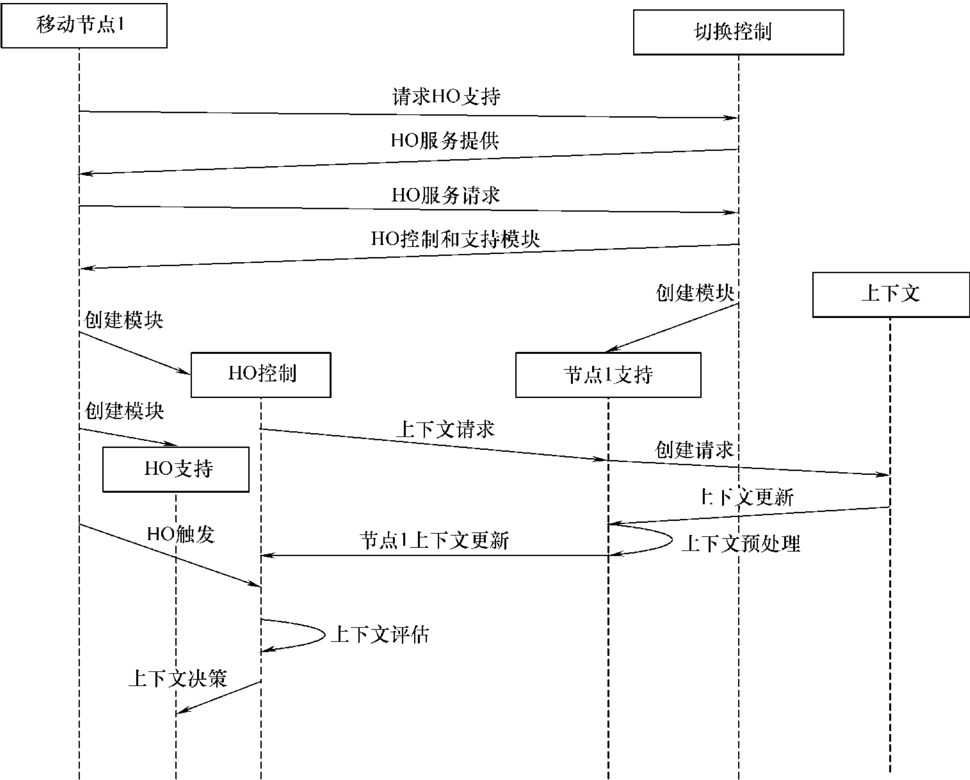


图 10-9　上下文感知切换中信令处理序列图

HDM 做出切换到哪个 AP 的决定之后，基于收集的上下文，所做的决策将发送给移动节点的移动性管理组件，以执行切换。在我们的方案中，移动性管理由移动 IP（Mobile IP）实现。

10.5.5　评估场景

本小节给出了一个负责上述最优化选择接入点的切换管理体系架构的评估。通过使用图 10-1 所示的场景和原型进行评估。

在评估过程中，考虑了两种类型的上下文信息：用户的位置、速度和路线，这些将取决于用户是如何移动的，以及 QoS。如果考虑了用户的位置上下文信息，那么所采取的切换将是沿着铁路线有更好的信号覆盖的网络或者是步行中有更高质量的网络。

10.5.6　实验和评估

定制的切换服务基于这样一个场景，即用户在火车上并通过移动节点观看视

频节目。

试验结果表明在有上下文信息的情况下，移动设备对于切换具备更大的控制权。对于不同的上下文环境使用正确的 CSA 和 CSAC 模块是很重要的。

当移动设备从 AP3 到 AP4 方向移动时，SCE 从移动设备的 GCCP 获得用户的位置信息。由于用户在火车上，移动设备上的 SCE 在用户和内核空间上安装了两个如图 10-8 所示的 CSAC 组件。

相应的 CSA 模块由本地的 SCE 安装于可编程网络节点。新安装的 CSA 将处理由 GCCP 搜集的关于铁路线附近 AP3 和 AP4 的信息。同时也是新安装的 CSAC 处理由 CSA 接收的信息并经过处理后发送给 CC。

为更好地进行试验，我们添加了一些考虑移动节点应用，如视频的具体需求的扩展。为此，我们将安装新的 CSAC 和 CSA 模块，用于处理包括节点位置信息以及接入网络负载在内的信息。基于搜集的上下文信息，CC 决定是否切换到 AP4，因为这个网络和 AP1 相比网络负荷较小。经过上述切换后，将获得更好的视频质量。

我们做了一些测量，用于评估提议的架构对网络和应用造成的影响，配置上下文感知服务所需时间以及搜集切换信息所需的时间。

在当前的原型中，在可编程网络节点的服务配置时间少于 1s，而对于移动节点将耗费几秒时间。上下文信息的搜集花费 1 ~ 2s 的时间，从 CSAC 请求切换信息到收到该信息为止，这个时间与移动设备和网络可编程节点之间的往返时间有关。在当前的原型中，包含 GCCP 的可编程节点也将评估 LIS 的功能。GCCP 预先收集上下文信息以及检索这些信息花费的时间非常短，因此在没有上下文信息就得做出切换决策的概率很小。

10.6　小结

我们方案的主要贡献在于定义了一个能够管理动态分布式上下文的通用架构，这些信息既可存储于应用层，也可能存储于网络层和链路层，同时它们将区分为特定应用的上下文信息以及原始通用的上下文信息。这种区分能够建立一套通用机制来收集、处理上下文信息，这些对应用是透明的。针对应用和网络服务进行的上下文信息调整由特定的适配器实现。在体系架构中实现了上下文管理架构。应用特定的适配器由服务配置体系架构实现安装。网络侧的适配器可以预先或根据需求进行安装，取决于服务配置体系架构的配置。

基于实现的原型系统，其实验结果显示了我们体系架构在处理不同上下文信息和配置的能力。我们的方案能够增强网络服务的性能（如切换），同时不会造

成移动网络性能的降低，而且定制化模块的使用也是很有效率的。

通过增强服务对周围环境的感知，其他相关的研究也已经提出了更多智能的网络服务过程。然而大多数方案仅仅考虑有限几个参数的切换研究上，例如无线接入技术类型、信号强度或接入网络的异构性。少数的方案定位于提供可扩展的收集、处理和分发上下文信息。

第 11 章　智能上下文

Matthias Wagner，Marko Luther and Massimo Paolucci

11.1　简介

在 Web 上执行复杂任务已经是我们现代生活中不可或缺的部分。移动服务的出现将为现有的 Web 服务带来更多的拓展和变化，并带来诸如定位信息的新功能。这些新型移动服务的成功展开很大程度上取决于它们在一个动态变化的环境中获得最优服务质量的能力。设备、移动应用和服务平台的上下文智能化对于管理不同类型的移动终端、个性化内容和服务以及缩小给定环境中大量服务集来说，都是相当有必要的。通过实施上下文智能化，这些移动服务和应用新增的能力对于灵活适配各种上下文环境以及最大化其价值都是很有用的。为此我们开发了语义网技术来实现上下文感知移动服务的智能化。我们的想法是把语义网构建在通用的计算环境之上，以便于表示和连接其上的内容、服务以及用户、设备、提供的能力和功能。在此基础上我们开发利用了 OWL 和相关的技术来为移动应用提供对语义网的访问。

本章的目标是对我们的智能上下文相关项目提供一个简明准确的综述和介绍。作为初始动机，我们展现了关于提供个性化 Web 服务方面的想法。集中于交互的不同阶段，看看协作发现算法如何真正改进所提供服务的质量。

11.2　研究原型

图 11-1 显示了我们目前正在实验室探索的与语义网相关的项目范围。在研究的分布图中，正在进行的活动可以根据其与 OWL 基本原理在表示能力和推理支持能力方面的关联程度进行分类。整个项目也可以映射到其他的应用维度，各维度代表面向语义支持上下文的表示和管理移动服务的各个活动。

特别强调地是，语义计算和语义网的真正成功与否将主要取决于 W3C 推荐的 OWL 是否能够获得大量支持，以及是否被应用于重大的行业级应用中。为此我们通过移动服务能够访问语义网的工具开发来开拓 OWL 的应用。同时我们的想法是将语义网建立在普通的计算环境之上，这样使得与内容和服务的显示和连结更加容易，也包括和用户、设备所能提供的能力和功能。在本章中我们将展示

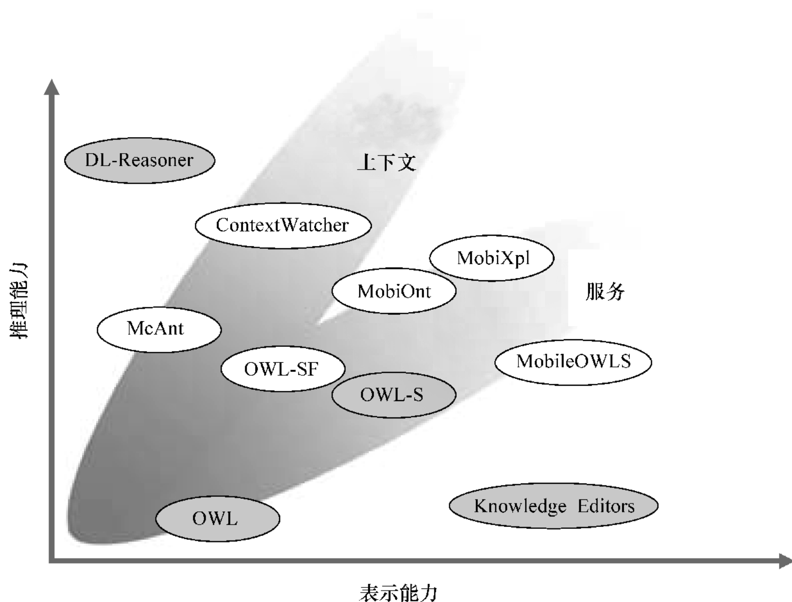


图 11-1　语义计算和语义网相关的项目

在之前的工作中所掌握的知识和经验，给出 OWL 及其工具改进的一些讨论。

11.2.1　OWL-SF

在设计通用的上下文感知系统时所面临的主要挑战，包括上下文信息的分布性以及提供服务和上下文信息的设备的多样性，这在我们开发的一个分布式语义服务框架 OWL-SF 过程中得到了展现。在这个框架中，W3C 的 OWL 被用于捕捉高层次的上下文内容。通过使用 OMG 的 SDO（Super Distributed Objects）技术来封装设备、传感器和其他实体，使用 REST（Representational State Transfer model）进行通信。通过研究一个使用增强的在线控制来实现智能呼叫中继的例子来评估整个框架。

1. 功能架构

如图 11-2 所示，OWL-SF 框架的功能架构可以分成两个基本块，即 OWL-SDO 和 DS（Deduction Servers）。一个系统可以具有多个这样两种类型的组件，它们可以在运行过程中动态地进行增加和删除。DS 基于从可访问的 SDO 收集到的结构信息进行推理支持，以及基于推论进行服务调用。相比较而言，OWL-SDO 实现了特定的程序逻辑，通过提供增强 OWL 功能的 SDO 接口封装底层的硬件和软件组件。通常两种类型的组件都可以提供同一种接口（例如 DS 可以支持 OWL-SDO 的接口或 OWL-SDO 可包含一个本地推理组件）。然而由于两者在概

念和逻辑上还是有区别，所以在体系架构中我们仍然进行区分。

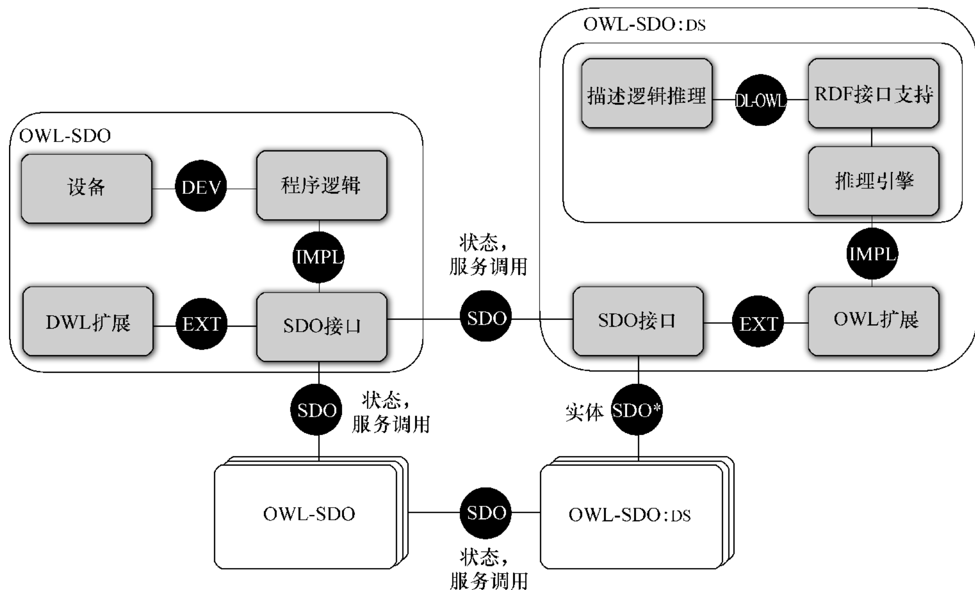


图 11-2　OWL-SF 功能体系架构

体系架构中有两种类型的参考点。一个是 SDO 参考点，表示 OWL-SDO 间以及 DS 和 OWL-SDO 之间的通信；而另一个就是 SDO*，负责建立 DS 间的链路。服务呼叫的实现以及在 SDO 内部状态的信息将通过 SDO 参考点进行交互，而 SDO* 参考点则用于交互各个单独的推理服务器信息。

2. 上下文感知的在线管理

基于 OWL-SF，将实现一个面向办公室环境的智能呼叫中继系统，并展现上下文感知的在线管理来实现智能呼叫中继。这个场景非常适用于这样的人群：他们都拥有移动电话，且在同一个办公室或会议室开会。依据一个人的位置、周围的人员、当前时间等，其环境信息可以进行分类并进一步用于确定其是否将接电话或者处于忙碌状态并且需要中继。

如图 11-3 所示，整个展示系统使用 Java 进行开发，并且具备丰富的 GUI 对人员、移动电话、房间等进行可视化。用户可以修改人员的上下文信息，移动电话也可以进行配置，通过蓝牙或 Parlay 进行连接。为简单起见，模拟环境中时间的变化将配置成两个可相互切换的时间状态：办公时间和午餐时间。类似地位置检测信息的获得通过在计算机特定端口上插拔存储块实现。

通过使用分类决定一个角色所处的场景以及电话的可接入性，图 11-4 展示了一个 OWL 概念分层的例子。例如一个人处于办公时间时，其所处的环境可以

分类为工作中；如果一个人正在工作同时处于开会中时，其所在位置可分类为会议室。

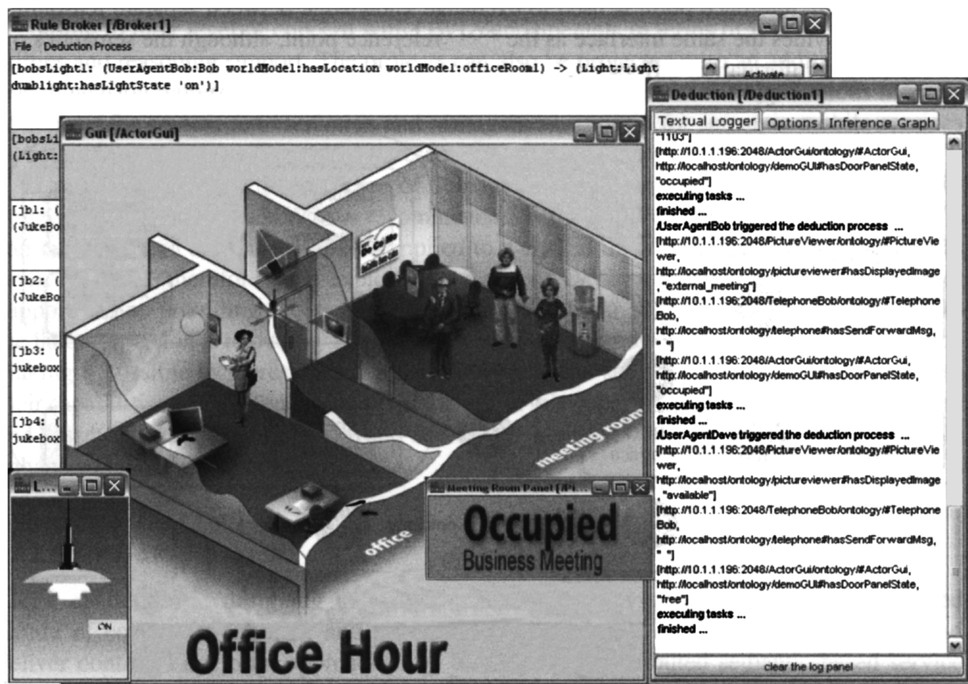


图 11-3　OWL-SF 模拟环境

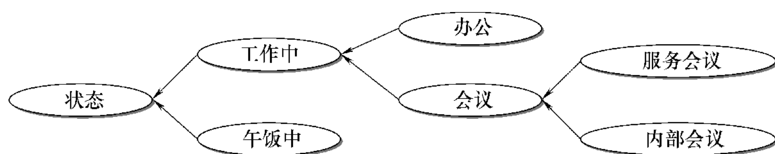


图 11-4　模拟实体划分

我们早期的研究显示通过结合 SDO 技术和 OWL 来创建一个完全分布式的语义服务环境是可行的。然而，我们需要综合考虑隐私和底层上下文推理机制来拓展我们的方法，这将带来新的挑战，例如传感器搜集上下文信息很难精确。

11.2.2　上下文感知器

在 IST 项目 MobiLife 中我们已经实现上下文感知器（见图 11-5），它是一个早期的基于语义的移动用户检测原型。该项目定位于为用户日常生活提供上下文

感知的服务。OWL 高层的上下文实体定义了基本的上下文目录和它们之间的关系，这些高层的上下文信息结构能够和语义有效整合，而且上下文元素的原子描述例如个人状态（如工作中、在家等）能够为对用户在线和虚拟位置实现推理的逻辑引擎直接使用。



图 11-5　ContextWatcher 移动终端

1. 上下文共享

上下文感知器使得人们对于共享上下文信息变得容易和顺畅。共享的上下文信息可以包括它们的位置信息或它们愿意和他人分享的任何信息，例如本地天气情况，甚至是一些像心情、经历之类的个人信息。应用的目标是为人们提供新方式来保持沟通联系，而不依靠传统的面对面方式。举个例子就是每个星期浏览朋友的在线相册。相片所蕴含的上下文信息对一个人的活动给了一个很形象的展现，当和朋友再次见面时便可以作为话题进行交流。上下文信息共享的例子包括：

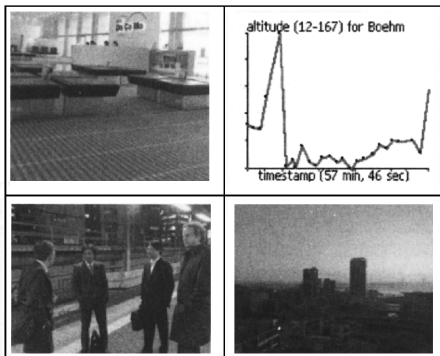
1）基于上下文增强的好友列表：这是两个人之间实时的上下文信息共享。共享前，两个人先得成为好友，并就共享哪些特定类型的上下文信息达成一致。在任何时候，好友都可以看见朋友在哪，他们正在阅读什么书，朋友所在地的天气怎样，当前的心情如何，他们和谁在一起。这些信息是以好友列表的形式展现的，这和即时消息应用中普遍使用的方法类似，但在上下文感知器中，它将在上下文菜单中提供位置和其他详细的上下文信息，如图 11-5 所示。此外好友可依据 Ad Hoc 连接等方式进行分组，这样在家的朋友可以很容易和在单位的朋友区分开来，能很容易地发送一个短消息给在单位的那些朋友说下班后去喝一杯。

2）个人上下文感知日志：在日志中既可以提供及时的上下文信息，也可以用人可阅读理解的方式提供上下文概况。好友列表的重要任务是显示一个好友的最后位置信息，在日志里，位置信息的时间序列需要组织成一个单一句子，例如“今天我从 Enschede 到 Amsterdam 然后返回”。这些信息可以由用户在日志里进行配置，基于检测到的上下文信息可以自动生成日志项。每个人都可以有这样一个每日行程信息的日志，包括访问过的地点、当地天气、遇见的人以及读过的书

等。这样一个日志对于住在远方的家属亲戚等来说还是很有吸引力的。图 11-6 展示了一个日志项例子。

星期一，2006.12.04
在东京的一天

今天的天气情况：在横滨的温度为6/11℃,天气晴朗比较凉爽，湿度56%，轻微东北风。
拍摄的照片如下：



今天经过的城市：横滨(4.5h),东京(19.1h)
今天遇过的好友：Wagner(21.5h),Furusem(10.6h)

图 11-6　ContextWatcher 博客的实例

上述这些上下文信息的共享方法都展现了通过创新的移动应用，保持和朋友、家人、同事之间的联系沟通是很容易的。惟一的要求就是用户需要一部具备上下文感知器功能的移动电话并随时开通。好友列表和日志自动更新，用户只需根据其需求进行配置即可。

2. 上下文感知器的主要特征

上下文感知器应用在安装上是模块化的，能够整合来自不同开发者的不同组件，在组件层次上是可动态配置安装和自动更新的。通过这种方式可以提供统一应用的多个版本，例如，轻量级、特定需求型、全功能型。该应用容易为新开发者进行扩展以及原有开发者进行维护。图 11-7 展现了相应的体系架构。

和上下文提供方所在网络的交互是通过 GPRS 或 UMTS 传送的。从本地传感器获得的数据将被推送到远程的上下文信息提供方，他们将处理这些信息并和授权用户共享信息。

1）增强：这包括向其他信息服务方咨询，获得上下文参数的更多信息。例如提供 EAN-13 编码对应的产品名称和类型，或者从 GPS 获得物理位置信息。

2）推理：通过对来自不同人员的上下文数据进行分析提供一些推理信息。例如，当检测到多个人员在同一个房间，有一些职员和一个领导，那么可以推理

他们正在开会。

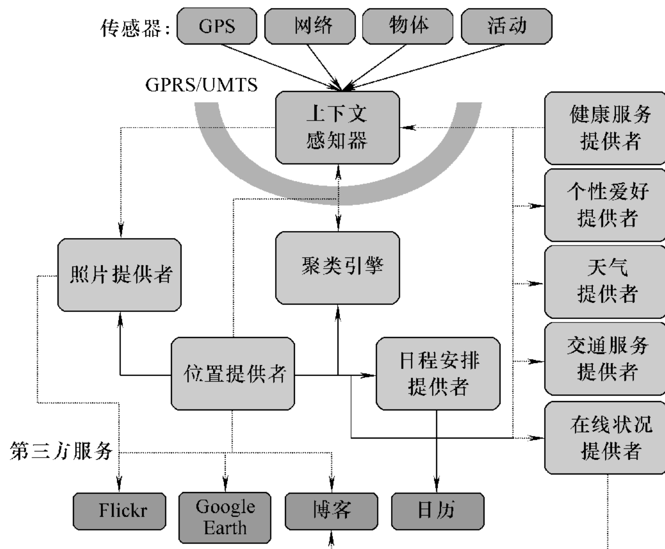


图 11-7　ContextWatcher 应用的体系架构

3）分享：以原始素材或增强处理的方式向授权用户提供上下文信息。例如给好友或第三方服务实体如日志和相册等。

例如，与位置服务提供方的交互如下：源位置数据的获得是通过 GPS 接收器（经度、纬度）或者手机的网络信息服务（cell id、网络、国家）。这些源数据定期被发送到位置服务提供方，他们可以通过一个大型的、独立于运营商的 cell-id 数据库，例如 http：//client338. lab. telin. nl：8080/wasp/jsp/CellStats. jsp，来分析 cell 数据，也可以通过 http：//www. mappoint. net 等来修正位置信息，并存储在数据库里。这些信息将被其他的组件所利用，包括分类引擎、相片提供方和天气信息提供方。

分类引擎逐个分析每个人的位置时间序列，试图找到其经常出现的地点，以便于将原始的位置数据转换成对用户或其好友有意义直观的数据。在谈话中，我们并不会提及绝对位置，而是说相对位置：办公室、Marko 的家或教堂附近等。这些相对位置信息容易被理解，在交流中能得到更好的使用。分类引擎每个晚上都对位置流信息进行分类，它能够扩展原有的分类或找到新的分类，并由上下文感知器提交给用户进行命名。用户的命名通常是一些位置实体，例如家、办公室、旅馆、运动场等。

照片提供者存储照相手机所拍摄的照片，并利用照片拍摄地点的环境信息自动为其添加分类标签、标题及描述。这些环境信息通过其他的上下文信息提供者

获得。通过这种方式我们可以根据自动记录的街道或城市、地理位置、运动方向和速度、当地所对应的分类名称、天气、周围人员等信息，给每张照片打上标签。

我们已经将照片上下文提供者与 Flickr（http://www.flickr.com）整合起来，这是当前最大的一个公共照片管理服务器，在那里上下文信息以标签的形式提交，描述文本将自动产生，例如，我和［Henk］以及［Bernd］在［Oulu］出差并在从［机场］到［夏令营］的路上拍了这张［Alexanderkatu］的照片，中括号内的所有信息都是自动生成的。这也就是说移动电话能够发送一个包含足够多说明的照片给远程的图像服务器，以确保他人能够从照片中找到他们感兴趣的信息。例如，通过浏览标签来区分在家拍的照片和在办公室拍的照片。

天气信息提供者提供城市的当前天气情况和天气预报。由于城市信息可以从位置服务提供者获得，因此不需要用户来指定在哪个城市，一次点击即可更新任何当前城市的天气信息。天气信息在上下文感知器中显示并存储下来，这样日志产生引擎就可以向天气信息提供者查询用户某一天所经历地区的天气情况了。

除此之外，还有很多功能可以实现，例如社会关系的管理、体育赛事实时数据、声音的记录和共享、条码识别等。同时在不久的将来还会更多的改进和发展。

11.2.3 McAnt

推理支持的一个前提条件是已经有相应的定性数据。我们可利用已经存在的高级别上下文信息，如存储在 PIM（personal information management，个人信息管理器）中的数据，对于从上下文数据低级别到高级别的处理不是必需的。

McAnt 原型能够支持基于来自不同地方定性信息的上下文推理实验。它阐述了已经存在的数据如何能够被推理实体所访问，以及衍生出的知识如何用于增强应用。McAnt（见图 11-8）已作为 Java 应用程序被开发出来，并带有原始 Mac OSCocoa 接口。此外，它还链接到后端推理引擎 RACER[12]、OSX 地址簿、日历应用及 MobiLife OWL 本体。

1. 检索定性信息

大多数用户会及时更新他们的电子地址簿并充分利用日历等应用。为此高层次的上下文信息通常是可得到的，也是可由 PIM 应用进行访问的。如果上述这些无法实现，那么将传感器采集的低级别数据映射到高级别的上下文信息将是必不可少的一部分。

如图 11-9 所示，存储在地址簿应用的信息通常不仅包括一个简单的联系信

息数据库，此外也提供描述联系实体之间关系的标签。根据这些关系的定义，可以对用户进行分类，如家人、同事等。类似地，使用日历管理器 iCal 管理的信息可以由 McAnt 进行访问和添加。

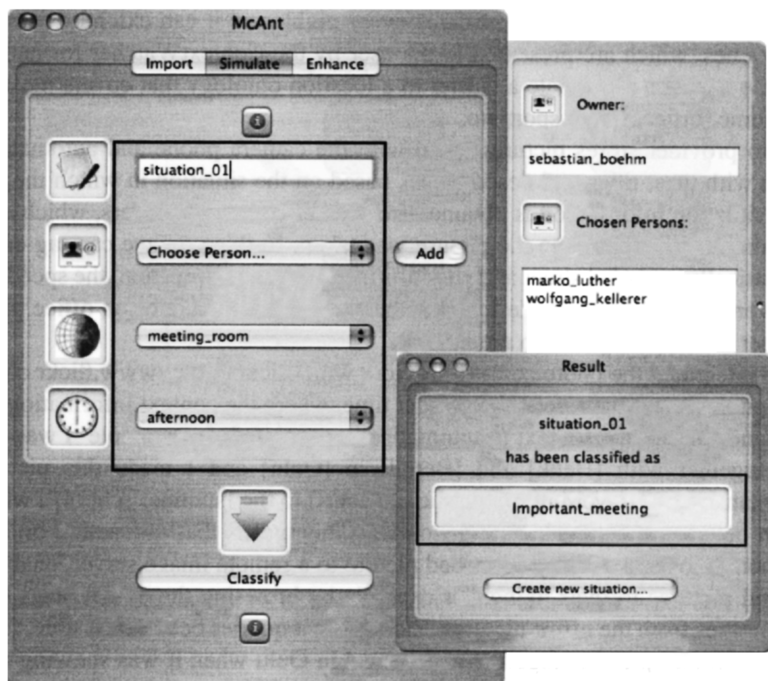


图 11-8　McAnt 模拟环境

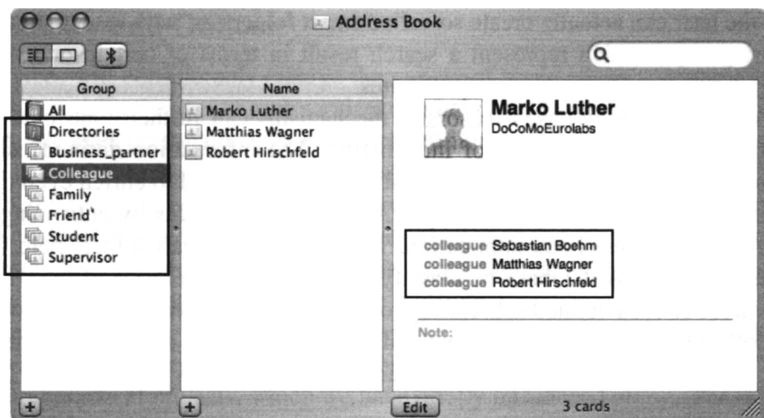


图 11-9　Mac OSX 地址簿

2. 场景模拟

由于能够导入高级别的个人上下文信息，McAnt具备模拟特定场景实例的能力。通过其简单的用户接口进行选择，例如在下拉菜单中选择参与的人员、位置及时间，就能够创建新的应用场景。

不仅仅是组织上下文信息，推理实体还衍生出新的知识信息。作为一个简单的例子，通过定义人员关系属性，将可具备自动完成人员角色划分的功能。

3. 应用增强

操作系统领域正在进行的开发，表明智能搜索引擎以及管理数据的智能方法将变得越来越重要和关键。例如苹果公司（Apple）最近已经发布了第四代桌面操作系统Mac OS X Tiger，包括一个叫Spotlight的搜索引擎。这种系统级别的搜索技术已经被整合到大量的应用中，包括Apple的标准PIM应用如地址簿。

推理机制可以进一步被应用于组织管理不同类型信息的智能方法中。使用逻辑推理得到的信息可以用于丰富应用。McAnt通过加强Apple的地址簿应用展示了这方面的应用。这些扩展的智能文件夹有助于按照和主人的关系来浏览地址簿数据库中的信息。

11.2.4 MobOWLS

MobiOWLS[13]是一个新项目，在这个项目中我们进一步扩展OWL-S[14]，以改进移动计算的服务描述。我们初始的调研集中在扩展OWL-S Profile，使其包含关键的QoS信息以及上下文信息。例如我们通过媒体等属性扩展OWL-S，这用于指定媒体的类型。

11.2.5 PERCI

移动设备和物理对象之间的交互得到了人们越来越多的关注，因为它以一个直观的方式从现实世界的物体中请求服务。我们已经发现有几种方案提供这种服务，它们中的大部分是私有的，且用于特定的应用领域或交互技术，没有提供通用的概念来对现实世界中实体所请求的服务进行描述。另一方面通用环境中有许多标签化和网络化的实体，如互联网环境。在项目PERCI（Pervasive Service Interaction）中，我们的目标是利用语义网技术来丰富标签化交互方法。特别地，我们研究语义服务框架（如OWL-S）是如何能够被拓展到面向通用服务交互场景的。

11.3 小结和展望

移动服务将大大拓展Web上的各种应用。我们讨论了如何通过可扩展的服

务描述机制、个性化机制、先进的服务发现和运行方法来有效使用这些增强服务。未来像 XML 加密等基本技术将和先进的标准和描述语言进行结合。我们已经建立了一个基于已有技术和研究成果的个人移动 Web 远景。

在我们的研究中，如在移动或通用服务环境下的上下文智能方向上，我们将语义网技术应用于之前介绍的多个项目中。这些项目要么集中在对基于 OWL 开发的基础支持上，要么集中在移动计算场景的服务和应用中。我们也阐述了所面临的一些挑战，以及我们在处理上下文信息中所获得的一些经验。然而尽管取得了一些进展，但仍然有许多富有难度的挑战在等待我们处理。特别地，在利用 OWL 过程中，我们发现了其在语言规范上以及工具的支持等方面还存在一些限制。

第 12 章　从个人移动性到移动个性化

Matthias Wagner

12.1　简介

新服务一直难以被用户接受，最普遍的原因是使用和访问新服务的难度较大。因此未来服务能否成功，关键在于身处特定环境和条件下用户是否能够随心所欲地方便地获得服务。在本章，我们将介绍移动个性化的概念，这个概念是建立在本书前面章节所介绍的许多概念和技术基础之上的。这个想法是允许移动用户能够根据个人喜好、用途以及不断变化的服务配置和服务类型来开发他们自己的在线个性化服务，以获取独特的主动行为。这种自适应个性化服务在本质上基于高级轮廓和个性化的概念、上下文感知计算以及灵活和不断演进的服务支撑中间件。通过实际的应用实例，结合详细的解释以及对潜在使能组件的合成，我们将尝试对预期的从个人移动到移动个性化的转变进行深入的剖析。

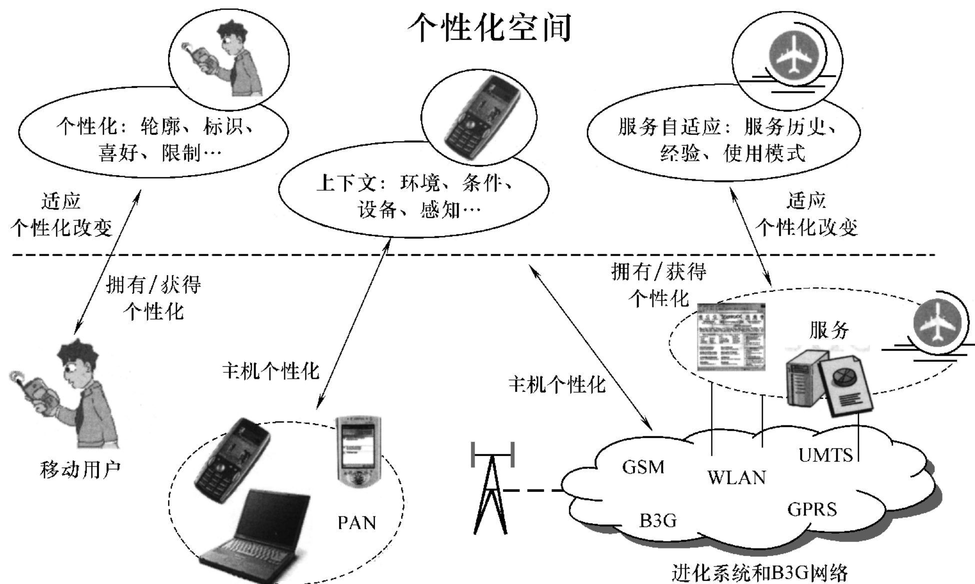


图 12-1　3G 以后系统中的移动个性化的愿景

假设 B3G 系统包含异构的接入网络，以方便向用户提供最好的可用移动连接。这些系统不仅仅用来集成多个网络平台，而且能够有力地促进更为丰富的服务和应用程序的开发。另一方面，随着网络复杂度地不断增加，我们仅仅能够对大多数终端用户将会很快面对的大量服务以及多种服务之间的结合方式有一个粗浅的印象。在访问以及使用新配置的服务中出现的问题一直是我们所提及的导致服务投入应用缓慢的原因。这个原因对于将来新出现的多媒体类型的服务和上下文感知的应用来说显得尤其重要。因此，提供用户感知和上下文感知的新服务将是非常重要的。

本章（以前在《Telenor Telektronikk》杂志上发表过）取材于我们正在研究的 B3G 系统中个性化服务供给支持的核心概念。在诸如 EURESCOM Project P1203（http：//www. eurescom. de）和 IST MobiLife（http：//www. ist- mobile. org）中，我们和欧洲运营商、设备商和厂商们一起确定了很多未来移动通信系统中的系统概念和应用领域。我们认为足够充分的个性化概念以及主动服务的宣传和采用可以有效地利用未来的服务。新的服务提供和部署的概念以及新的服务算法都会吸引众多消费者并且给移动通信带来超越传统语音应用的巨大市场。接下来，将概述我们提出的演进的虚拟个性化概念，它并不仅限于用户，也适用于移动系统的其他实体。这些能够实现自适应的个性化服务，能够主动开发自己并且扩展成一个“智能化空间”（见图 12-1）。随着服务部署和应用的不断发展，这样一个“移动个性化”允许不同的用户根据他们的个人需求开发自己的个性化轮廓以及典型的使用特性，并且也允许他们提供服务来获得惟一的主动行为。

12. 2　未来用户配置和个性化

个性化服务和应用是未来移动通信系统最显著和最需要的功能。它们能够使用户从不断增长的、多样化的移动服务中对服务进行选择并根据用户自身的需求进行服务调整中获益。以商业旅行者为例，我们已经通过应用场景对未来移动通信系统中的主要个性化问题进行了概述。下面我们将重温这些场景，尤其是针对移动个性化演进的场景。

假设我们有一个名叫迈克尔的用户，他打算从波士顿到巴黎进行一次商业旅行。这此旅行中包含很多不同的步骤和许多任务。要完成一次充分的旅行准备是非常复杂的，并且手动地找到足够的服务也是非常耗时的。况且，对服务的许多需求和偏好既不能满足迈克尔的需求也非常地令人感到乏味。为了简单起见，我们假设迈克尔本次旅行的个人计划任务仅仅包括如下几项：

1）安排好必要的行程；

2）开会地点的导引；

3）与不断改变的环境相适应的服务和用户设备。

显然，迈克尔的行程将会以从波士顿到巴黎的航班开始。举个例子，通过交互式的Web页面和一个移动终端，航班订票代理可以根据迈克尔的个人喜好购买一张机票。因此，需要搜集所有可用的飞行服务。这可以通过旅行门户网站或者旅行代理来完成。不论哪种方式，直觉上一个灵活的办法就是需要支持对用户喜好的建模和表述。迈克尔的用户配置可能包括对出发和到达日期，对机票的舱位级别（比如说商务舱）的硬性限制。另外，还可能有一些软限制，比如对航空公司的喜好（我更喜欢法国航空公司，而不是Delta航空公司）、飞行类型（直达）以及航线（越快越好）。迈克尔的所有喜好都存储在他的个人配置里面，而该配置可以存在他的个人设备或者使用服务（比如说3GPP http：//www.3gpp.org）分布式地存储在网络中。

如同订票一样，迈克尔的旅行代理可以在到达方机场安排一辆租用的汽车。成功地订好航班后，迈克尔最终到达了旅行地。他直接去机场汽车租赁中心取预订好的汽车，并通过手机进行身份确认和授权。他的手机透明并且自动地对可用的服务进行发现。车内的所有装置与手机同步地对车内的镜子、座位和温度根据迈克尔的喜好进行调整。除了他喜好的驾驶装置外，迈克尔的手机还发现了汽车内置的导航系统。导航系统马上给出会议地址和相关地图。通过使用本地交通信息，导航系统选择了一条路径并给出了预计到达时间。由于车中系统提示目前离开会时间还早，迈克尔决定在本地提取一些现金。他接着访问了ATM的定位服务，并通过它获得了离他当前位置最近的ATM机的位置，并且以最低的手续费支取了现金。一旦迈克尔选择了一台ATM，导航服务将调整路径并且和其他服务找到距离当前位置最近的停车场。

最后，迈克尔通过指引到达了会议现场。通过他的环境感知的通信环境，除非发生紧急事件，他不喜欢在会议期间被打扰。因此，迈克尔的通信设备装置在会议开始时自动进行调整。这并不需要显式地切换到另一个设备视图。如果没有任何一个可用的设备视图能满足条件需求，一个外部的配置参数会暂时地传输并仔细检测紧急设置。在会议中，迈克尔能够把他的视频流从手提设备传到嵌入式的显示屏以及会议室的会议系统。除此之外，他也能通过设备发现其他的诸如最近的打印机和视频转码器等服务。所有的这些设备都根据迈克尔的个人配置进行初始化。由于一个重要的参会人员被临时叫走，该会议不得不延期到当天的晚些时候。同时，迈克尔想同该公司的其他人员进行会谈。一个可用的基于WLAN的调度服务可以进行短期的安排。它对所有人员和相关管理信息（比如房间号、联系电话等）进行了详细的罗列，而这些信息都来源于用户的当前配置。

12.3　新移动生活

在将来的移动平台上需要考虑很多系统以及具体的实现细节。这些都是个性化服务提供以及移动个性化实现所需要的。我们尤其需要考虑如下的需求：

1）通过高级配置对用户进行建模；

2）对环境的感知和建模；

3）支持以用户为中心的服务发现和服务选择；

4）处理建模信息并支持服务执行和服务适应；

5）灵活的支持服务的中间件，它能够传送配置和服务以及个性化的服务组件。

接下来，我们将继续讨论移动个性化的主要构成。

12.3.1　高级个性化概念

在我们的场景里，一个用户的“个性化”折射为与他相关的个人配置并且包含配置向其他用户、网络节点和服务提供商的传递。用户建模和配置超越了设备的独立性，它能够覆盖用户的喜好和愿望，因此能够支持这样的虚拟化移动个性化。最近建立了许多相应的配置标准来描述服务传递的上下文：W3C 创立的组合能力/喜好配置（CC/PP）标准，由 OMA 创立的用户代理配置（UAProf）标准，以及由 3GPP 提出的通用用户配置标准。他们定义了一个基于 XML 和 RDF 的框架来解决设备独立访问的通用需求问题。虽然他们提供了有关配置信息的互操作的基础，然而现在的配置语言还是不能满足高级配置的应用需求。

如同在场景中描述的，基于语义和合作式服务发现及选择是主动服务的组成部分，也就是说，用户的需求和愿望被认定为复杂的任务，通常情况下被进一步划分为简单的子任务来更好地满足用户的上下文和环境。对于用户配置语言来说，需要从很大程度上借鉴于知识工程和数据库领域的相关知识，在这些领域中配置元素的分类和组织通常理解为一种方法或者是本体。我们赞成借鉴语义网的知识来设计未来的个性化配置语言。因为在语义网中，内容描述层面有一个相似的特性，那就是在 XML 上，RDF 提供了一个简单但是一致的结构来为不同的 Web 本体语言提供基本的语义描述和基础。对于主动化的服务，一个用户任务的子目标不得不进行更为深入的研究和划分并最终匹配到足够的服务。在此过程中，根据用户的个人喜好，可以借助高级发现和选择机制来搜寻用户的服务环境。

12.3.2　上下文感知计算和管理

上下文感知是服务的基本属性，它能够访问、解释和操作环境知识并对服务

行为进行自适应。为了使得只有上下文感知服务能够开发他们自己的个人属性，我们开发了一个上下文管理框架来方便基于 Web 服务框架的上下文获取、管理和处理。

这个框架在 MobiLife 项目中开发，并且在 MobiLife 上下文管理框架（CMF）中得以应用。该框架支持用户上下文的发现和交换，也支持关于上下文信息的推理。它的目的是支持上下文信息能从一个提供者很容易地流向多个上下文消费者，同时也能够从多个提供者流向一个消费者，从而建立一个智能的提供者体制，并创造高层次的条件信息。这个高层次的信息的建立基于来自异构源头的上下文信息的若干比特。这些异构包括异构的空间维度、语法、语义、安全、协议以及上下文质量。CMF 架构的主要任务如下：

1）支持上下文提供者的发现；

2）标准化提供者和消费者的上下文交换；

3）通过允许推理组件来支持容易的上下文推理，因此一个推荐者可以以即插即用的方式加入到应用中；

4）支持不同上下文提供者的构建。

MobiLife CMF 是一组组件，它们在运行时动态地互联，并一起采用感知和解释机制来提供相关的上下文信息。一个上下文提供者的具体实现方式可以是简单的封装一个探测器来为单用户服务，也可以是构建一个推理器来实现多个上下文提供者的合成来为多个用户服务。CMF 已经被成功地应用来实现和支持不同的 MobiLife 应用程序，包括上下文感知器。

12.3.3 灵活的服务支撑中间件及其演进

服务支撑中间件需要具备高效的会话管理能力，包括配置管理和服务移动性。作为服务支撑中间件的核心部件，服务会话控制需要提供高级的服务供给功能。我们把应用层信令看作协调机制，它支持行动实体间的信息交换。为了支持高级的个性化概念，我们支持基于代理服务器的服务信令方式。在此方法中，配置管理、服务会话管理和资源控制都实现在分离的交互的代理服务器群组中。这个概念事实上是对基于偏好的会话管理的一个增强。基于偏好的会话管理利用了在代理服务器（比如说网络边缘节点）中的配置信息。

另外，未来系统需要对用户需求和应急服务有足够灵活性。到目前为止，基本上所有的服务架构都处于层次系统结构必须的功能层次上。另外，我们已经在中间件层次上描述了在 B3G 系统中支持自适应和演进的架构。这样一个自适应的环境可以通过灵活的功能组件（比如上下文管理或者服务选择器）来实现高效的服务执行。而在所有系统层次上的可编程平台构成了这个组件管理的基础。

服务的自适应和可编程性不应该仅仅局限在当前许多系统中所看到的内容自

适应方面。相反地，它应该包括行为（服务逻辑）和服务交互以及信令的修改。动态服务的自适应是如何受益于诸如面向领域编程的动态可编程环境的将在后面由 Hirschfeld 等[15,16]进行详细介绍。

12.4　主动服务发现的偏好模式

使用基于 Web 的服务已经成了我们日常生活的一部分。语义网技术以及通用和移动访问互联网服务技术的出现能够增加当前 Web 服务的宽度并且能够提供额外的诸如基于知识的、位置感知的或者是上下文感知的一些附加信息。另一方面，到现在为止，有关语义服务框架中的移动计算方面的工作还非常少。然而，许多语义网服务发现和构建的许多工作大多关注于服务的功能、上下文信息、个人喜好以及越来越普遍的个性化，而这些都是移动服务计算领域的经典难题和挑战。为了管理日趋增长的移动服务，语义网服务标准需要在本质上支持开发者和用户的需要，比如在特定的条件和上下文环境下服务的发现和选择问题。

在早期的研究中，我们实现了 MobiOnt 和 MobiXpl 两个语义工具箱在语义网中来研究以用户为中心的移动服务。我们希望充分利用未来在有限终端设备的复杂网络供给来处理移动环境下的个性化服务发现的需求。MobiOnt 和 MobiXpl 是早期实现并作为 Protégé 知识工作床（http：//protege. stanfor. edu）的插件以及基于 Java 终端的两个原型。MobiXpl 能够模仿许多商业的手机，而 MobiOnt 封装了许多核心的基于偏好的生成机制。在实现方式上，MobiOnt 是作为一个中心的网络组件实现的，而 MobiXpl 是作为运行在一个手机上的基于 Java 的客户端实现的。

12.4.1　具体应用场景

我们现在来考虑一个以前发表过的关于移动互联网广播场景例子的扩展。互联网广播在过去的几年中发展迅速，它拥有大批的网络广播站和用户，例如网站：http：//www. radio- locator. com/。在这个例子中，对于个性化的内容访问在各种用户的个人需求中显得尤为重要。对于我们的用例来说，我们已经将互联网广播站建成具有多个用户特征的 Web 服务。广播信道采用互联网广播本体来进行描述（图 12-3 给出了这个本体的一部分），这个本体由许多描述和分类的 Web 广播服务的概念组成，它包括程序格式、音频格式特点以及基于时间的流音频分类等。这个服务本体将被用于基于偏好的服务发现。

12.4.2　用户偏好

在服务本体的浏览过程中，可以选择与用户密切相关的服务概念，并组合成偏好。在我们的偏好框架中，直接使用这些经过排序的功能，而不管其质量或者

排序值。用户偏好可以解释为与特定对象语义相关的特殊关系。比如说对象A（或者类A）优于另一个对象B（或者类B）（比方说在音乐频道和新闻频道中我更喜欢前者）。偏好意味着一些限制，一方面服务必须要满足需求，另一方面即使没有任何的偏好能够满足，也必须有一个最终的匹配存在。为了管理多个用户偏好，可以通过偏好合成器来将多个基本偏好合成为一个复杂的偏好。

图12-2给出了一个在广播场景中合成偏好的例子。在这个例子中，用户通常更喜欢来自于欧洲的广播节目，而不是来自于日本或者是美洲。然而，后两个选择也许在某些情况下更加优先于其他选择。另外，鉴于它的播放器的能力，它更倾向于MP3编码的节目，而不是Real编码的。并且，它认为所有的基本偏好对于它来说同等重要。

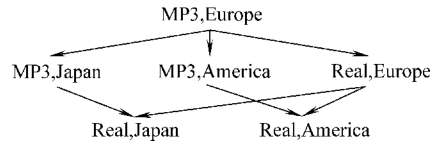


图12-2　一个面向用户定义偏好的排序

12.4.3　协同服务发现

用户偏好在构建过程中需要定义服务请求，并最终映射到底层的服务本体。MobiOnt（见图12-3）因此通过多种不同的策略实现了一个灵活的服务发现算法。服务发现的目的是为了从本体中检索出最能匹配用户喜好的服务实例。

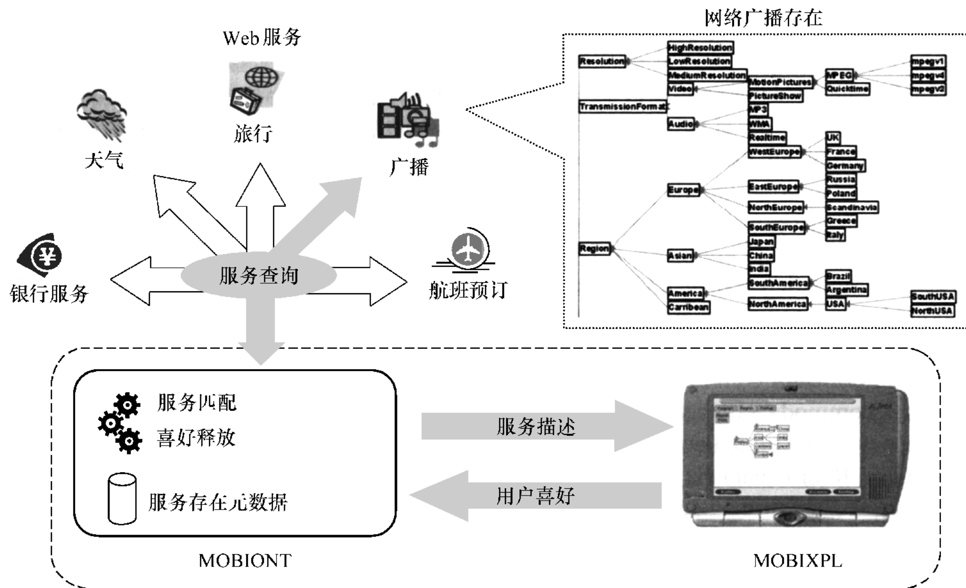


图12-3　MobiOnt/MobiXpl-一个基于语义的移动服务的测试床

实现的基于偏好的服务匹配是通过协同的方式来进行的：如果对一个广播电台的最佳匹配失败了，那么最初的查询就将沿着匹配的路径一直查找下去直到出现一个次优匹配。因此，在我们的例子中，如果一个 MP3 编码的欧洲节目查询不到，下一个查询的步骤将试图查询日本或者是美国的 MP3 编码的节目，或者是 Real 编码的欧洲节目。如果这两个次优的选择仍然不能满足，那么任何其他的节目都将是最终查询结果。更进一步的实现和应用领域以及选择本体浏览和喜好的构建和映射，都将在后面由 Balke 和 Wagner[8,20] 作进一步介绍。

12.4.4　MobiXpl——面向基于偏好发现的用户界面

互联网广播本体的部分通过 MobiXpl 这个图形化的终端较好地展现给了我们的终端用户（见图 12-4）。MobiXpl 模拟了很多不同的移动终端，并且具有一个移动本体浏览器以及一个针对用户喜好的用户界面。这个想法是根据用户的经验级别和用户轮廓来展现选择的概念和子本体。当本体被浏览的时候，与用户相关的概念将被选择并最终形成用户偏好。接下来，这些偏好将用来在服务发现的过程中来实现协作式工作：如果对一个广播电台的最佳匹配失败，最初的查询将沿着路径一直找下去直到找到次优匹配。所有的应用领域以及选择性的本体浏览，以及偏好的构建和映射，都完全是由 Wagner 等人[18] 研发的。

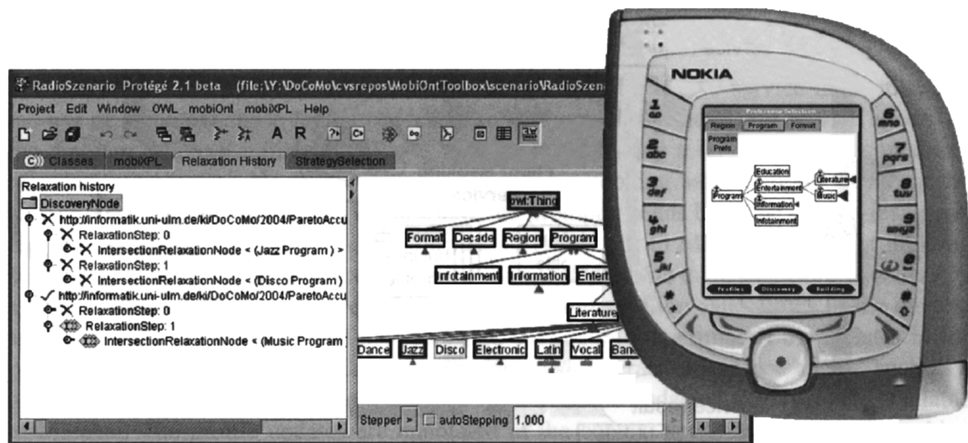


图 12-4　MobiXpl 在移动设备上研究基于本体的服务目录

12.5　面向移动个性化

用一种最直接和本能的方法来表达用户偏好，愿望和厌恶对于移动服务的灵活性提供是非常关键的。根据我们确定的移动个性化愿景，我们开发了相关的概

念和方法来提供灵活的用户偏好和服务的匹配机制，它们是：

1）一个基于用户喜好的服务间的协商和交互算法；

2）一套用于服务描述与服务用户模式的交互机制；

3）一个在服务目录中使用基于喜好的匹配机制的早期原型。

根据用户偏好对服务进行检索的最优匹配失败的情况下，通过协同方式可以达到次优匹配。这些概括的方法假设用户偏好的建模都是有其硬限制和软限制的。硬限制是在服务发现和选择过程中规定用户喜好是必须要匹配的，而软限制则仅代表了用户轮廓或者是需求的部分，这些在匹配过程中相对限制较少。我们的配置概念由表达用户组喜好和典型服务触发模式等组成。例如，一个旅游方面较为普遍的偏好是每个人都希望尽可能短的旅途时间（也就是说，从出发到到达的时间要尽可能短）。这些随着用户需求的变化而演进的模式在移动个性化中扮演着非常重要的角色。如图12-5所示，用户偏好被传递到服务提供者来进行服务匹配和执行。如果在个人配置和服务的默认配置之间没有匹配，两个基本且互斥的解决策略可以使用（或者可以结合使用）：一是协同式的服务执行可以用来暂时满足用户的偏好，直到一个匹配出现；另一个则是用户可以求助于一个典型的组用户配置，例如商务用户等。这样就可以从这个组用户配置中获取有益的信息以丰富用户的需求。

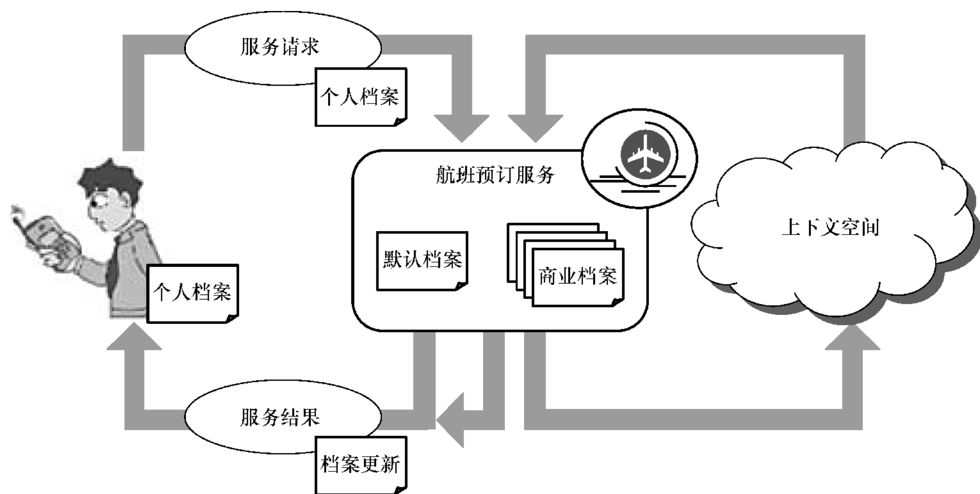


图12-5　协同服务执行和上下文空间

除了一次性应用，配置数据的生存周期在移动个性化中非常重要。比如，当一个智能代理在第一次为我们的主人公Michael在波士顿机场预留了一个汽车位，它会根据一些基本喜好来完成这个工作。然而，当再次访问波士顿时，迈克

尔的用户配置将会已经包含一则有关服务的使用历史纪录。那么，图 12-5 的左半部分对这个情况使用用户配置或者使用历史进行了描述，当然也许是其他的用户历史，比如来自于迈克尔的同学或者与他有着同样喜好的用户。在这个例子当中，迈克尔的租车代理将会根据过往一些历史纪录（比如车的破毁或者是技术问题等）来主动与车内导航系统进行协调和处理。

在图 12-5 中，我们给出了一个更加详细的自适应服务交互和使用上下文空间来进行智能代理间的互操作的场景。在图中我们可以看到，迈克尔使用用户请求来初始化航班预定服务。在收到请求后，航班预定服务将处理请求并马上反馈请求和更新配置信息。除了发送服务结果给迈克尔外，航班预定服务通过改写更新的配置来标注上下文空间。与迈克尔有同样兴趣的服务代理使用具有互操作能力的上下文空间来寻找批注后的空间，并使用逻辑确认这个更新是否能用来作为个性化的资料。通过代理间的交互，服务代理（比如一群订车位服务代理）推理得出需要更新额外的配置来完善迈克尔的个人用户配置。在上下文空间中更新的配置被标注出来，因此航班预定服务被通知异步地传送到迈克尔处。

12.6　结论和展望

面向 B3G 系统和服务的演进面临着不断增长的用户以及技术需求的挑战。我们认为在新服务的使用和访问中出现的问题在过去也曾使得服务的应用受到影响。因此，用户们可能并不能完全地理解新兴的服务以及从未来应用中可能获得的收益。在本章中，我们认为用户的接受以及服务的简便实用是即将带来的电信系统最为成功的因素，因此我们展示了我们的愿景以及面向支持高级个性化概念（也就是移动个性化）的第一步。

移动个性化是以个性服务、用户喜好以及不断随着用户环境和上下文变化的服务轮廓为特征的。这个个性化概念的主要组成部分是高级个性化概念、上下文感知的计算，以及高度灵活和演进的系统架构。我们未来工作的重点将研究上下文管理的通用方法、面向扩展语义描述（对服务、用户和设备等的描述）的服务发现和选择技术等。

第 13 章　主动服务走向现实

Hendrik Berndt

在本书中我们介绍了构建一种前所未有的新服务提供方法学、解决方案以及富有前景的深入视角。通过开发 4G 移动环境感知服务，我们向一个新的服务空间迈进。新的服务空间的特征不仅仅是我们在 Web2.0 新兴阶段看到的大量新应用，更主要的是服务语义和含义的增强。本体论是一个有效的理解服务行为模式的工具，而这些行为模式不仅表现在服务本身，也表现在服务的互联上。因此，服务可以在任何复杂的层次上进行组合。在本书中，那些对复杂环境具有适应能力并且能够根据用户兴趣极大个性化的应用所需要的构件都是组合使用的。这些组件的组合方式如图 13-1 所示。

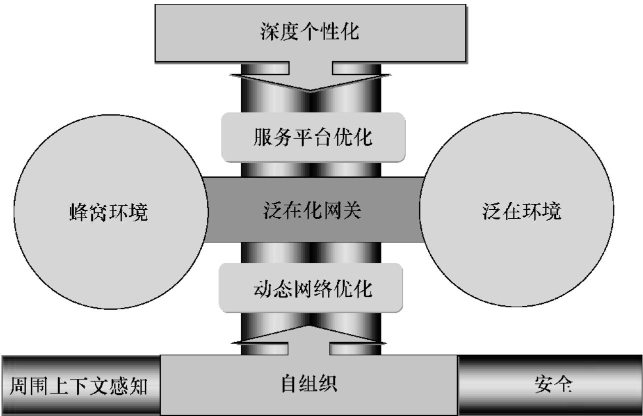


图 13-1　主动服务的构成组件

本书还概括了适合用户实际需求的高级服务发现方法。基于智能推理组件，在服务发出阶段，通过逐渐地替代用户的偏好以及放宽限制条件，可以考虑潜在的信息并解决冲突。随着服务的不断增长，用户指引也很自然地需要得到满足。移动运营商和服务提供商可以轻而易举地获得用户模式，因为他们非常清楚服务是如何在大范围内提供以及顾客最欣赏其中的哪些特性。这些知识使得他们可以创建用户指引以促进服务更便捷地被使用。这就是我们的主动服务最大的亮点。一个主动获取用户意图以及资源并加以考虑的服

务提供环境是非常有益的，并且极大地简化了生活。一个服务越能理解时间限制、位置、社会设置、服务兴趣、可用的设备等，它越能更好地主动调整配置，从而更好地服务用户。

在这本书中，我们展示了服务是如何组合起来运行在一个泛在的网络环境中，并从而扩展到应用于现实世界，其中包括所有具有通信能力的人造组件。信息社会中越来越多的近场通信服务应用于生活的方方面面以及互联网的各个层面。一般来说，随着技术的不断发展，用户能够获得越来越多的使用能力。他们的角色将会从商业角色发生很多转变，比如从用户到内容甚至服务提供商，从考虑安全和隐私的需求设置转向可靠和服务质量的需求。

有了感知环境的服务之后，很容易从邻近的朋友和特殊事件中获得匹配公众兴趣的社会组件。这为构建一个巨大的社会网络服务集合铺平了道路。我们看到了从传统基于位置的服务到上下文感知并进而进化到社会感知的移动服务的迁移，所有的感知层次都是相互构建和依赖的。这个面向社会网络的趋势如图 13-2 所示。

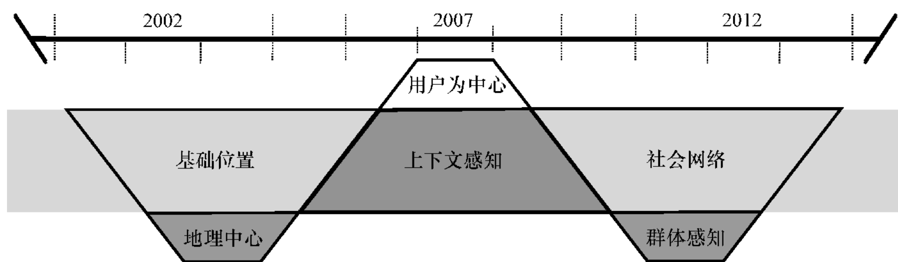


图 13-2　面向社会网络的服务发展趋势

享有一个公共视角的用户可以互相寻找、新建节点组织、交换信息、创建新的服务和组织事件。一方面，日常生活日志的自我呈现以及基于社区的服务需要这样的环境保护；另一方面，它提供了社会网络潜在性的一个侧面。

上述所有的服务和应用都是服务空间的一部分。我们使用图 13-3 描述的服务空间来总结本书的内容。它正在走向现实生活，并提供许多有意义的嵌入在现实环境和条件中的服务。

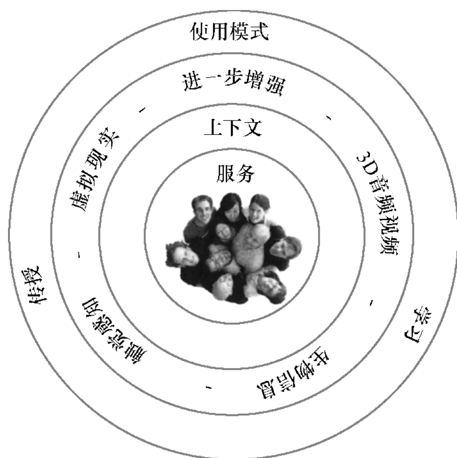


图 13-3　服务空间的愿景

在不远的将来，使用多种虚拟现实、生物信息、传感机制以及到目前为止没有用过的信息资源都能增强服务。本书另一个层次关注用户指引，基于使用模式的指导和学习，它包括基于资源的对服务功能的有效使用也对用户是可用的。

我希望尽快在服务空间看到你们!!!

附录 参考文献及延伸阅读

第1章 4G移动新架构

参考文献

[1] Inoue, Y., Lapierre, M. and Mossotto, C. (1999) *The TINA Book, A Cooperative Solution for a Competitive World.* Prentice Hall Europe.
[2] Prehofer, C., Kellerer, W., Hirschfeld, R., Berndt, H. and Kawamura, K. (2002) 'An Architecture Supporting Adaptation and Evolution in Fourth Generation Mobile Communication Systems' *Journal of Communications and Networks* **4**(4).
[3] Niebert, Norbert (2007) *Ambient Networks: Co-operative Mobile Networking for the Wireless World.* John Wiley & Sons, Ltd.

第2章 移动通信网络

参考文献

[1] Manner, J. and Kojo, M. (eds) (2004) 'Mobility Related Terminology' RFC3753.
[2] Fenner, B., Handley, M., Holbrook, H. and Kouvelas, I. (2006) 'Protocol Independent Multicast – Sparse Mode (PIM-SM): Protocol Specification (Revised)' RFC4601.
[3] Adams, A., Nicholas, J. and Siadak, W. (2005) Protocol Independent Multicast – Dense Mode (PIM-DM): Protocol Specification (Revised)' RFC3973.
[4] 3rd Generation Partnership Project (2002) '3GPP TR 23.846 v6.1.0, Technical Specification Group Services and System Aspects, Multimedia/Multicast Service, Architecture and Functional Description, (Release 6)' December 2002.
[5] Kempf, J. and Austein, R. (eds) (2004) 'The Rise of the Middle and the Future of End-to-End: Reflections on the Evolution of the Internet Architecture' RFC3724.
[6] Clark, D.D., Wroclawski, J., Sollins, K. and Braden, R. (2002) 'Tussle in Cyberspace: Defining Tomorrow's Internet', *Proceedings of the ACM Sigcomm 2002*, Pittsburgh, PA.
[7] Perkins, C. (ed.) (2002) 'IP Mobility Support for IPv4' RFC3344.
[8] Johnson, D. *et al.* (2004) 'Mobility Support in IPv6' RFC3775.
[9] Campbell, A.T., Gomez, J., Kim, S., Wan, C.Y. and Turanyi, Z.R. (2002) 'Comparison of IP Micromobility Protocols' *IEEE Wireless Communications* **3**(1), 72–82.
[10] Yumiba, H., Imai, K. and Yabusaki, M. (2001) 'IP-Based IMT Network Platform', *IEEE Personal Communications Magazine* **8**(5), 18–23.
[11] Host Identity Protocol (hip), IETF Working Group, http://www.ietf.org/html.charters/hip-charter.html.
[12] Braden, R., Berson, S., Herzog, S., Jamin, S. and Zhang, L. (1997) 'Resource ReSerVation Protocol (RSVP) – Version 1' RFC2205 (Standard), IETF.
[13] Davie, Bruce, Charny, Anna, Bennet, Jon, Benson, Kent, Le Boudec, Jean-Yves, Courtney, William, Davari, Shahram, Firoiu, Victor and Stiliadis, Dimitrios (2002) 'An Expedited Forwarding PHB.' RFC3246 (Standard), IETF.
[14] Heinanen, J., Baker, F., Weiss, W. and Wroclawski, J. (1999) 'Assured Forwarding PHB group' RFC2597 (Standard), IETF.
[15] NSIS, IETF Next Steps for Signalling (nsis) Working Group, http://www.ietf.org/html.charters/nsis-charter.html.
[16] Zhi-Li Zhang, Zhenhai Duan, Lixin Gao, Yiwei Thomas Hou (2000) 'Decoupling QoS Control from Core Routers: a Novel Bandwidth Broker Architecture for Scalable Support of Guaranteed Services' *ACM SIGCOMM Computer Communication Review, Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, **30**(4).
[17] Nichols, K., Jacobson, V. and Zhang, L. (1999) 'A Two-Bit Differentiated Services Architecture for the Internet' RFC2638 (Informational), IETF.

[18] Armitage, Grenville J. (2003) 'Revisiting IP QoS: Why Do We Care, What Have We Learned?' ACM SIGCOMM 2003 RIPQOS workshop report, *ACM SIGCOMM Computer Communication Review* **33**(5).
[19] Crowcroft, Jon, Hand, Steven, Mortier, Richard, Roscoe, Timothy and Warfield, Andrew (2003) 'QoS's Downfall: At the Bottom, or Not At All!' *Proceedings of the ACM SIGCOMM workshop on Revisiting IP QoS: What Have We Learned, Why Do We Care?* 25–27 August 2003, Karlsruhe, Germany.
[20] Manner, J. *et al.* (2002) 'Evaluation of Mobility and QoS Interaction,' *Computer Networks* **38**(February).
[21] Hillebrand, J., Prehofer, C., Bless, R. and Zitterbart, M. (2004) 'Quality-of-Service Signaling for Next-Generation IP-based Mobile Networks' *IEEE Communications Magazine*, June.
[22] Weiser, Mark (1991) 'The Computer for the Twenty-First Century' *Scientific American*. September 1991, pp. 94–104.
[23] Tschopp, D., Diggavi, S., Grossglauser, M. and Widmer, J. (2007) 'Robust Geo-routing based on Embeddings of Dynamic Wireless Networks, *Proceedings of IEEE Infocom, Anchorage, Alaska, USA* May 2007.
[24] Akyildiz, I.F., Wang, X. and Wang, W. (2005) 'Wireless Mesh Networks: A Survey' *Computer Networks Journal* **47**, 445–487.
[25] Gupta, P. and Kumar, P.R. (2000) 'The Capacity of Wireless Networks' *IEEE Transactions on Information Theory* **46**(2), 388–404.
[26] Bruno, R., Conti, M. and Gregori, E. (2005) 'Mesh Networks: Commodity Multihop ad hoc Networks' *IEEE Communications Magazine* **43**(3) 123–131.
[27] Niebert, N., Schieder, A., Abramowicz, H., Malmgren, G., Sachs, J., Horn, U., Prehofer, Ch. and Karl, H. (2004) 'Ambient Networks: An Architecture for Communication Networks beyond 3G' *IEEE Wireless Communications*, April 2004.
[28] Prehofer, C., Kellerer, W., Hirschfeld, R., Berndt, H. and Kawamura, K. (2002) 'An Architecture Supporting Adaptation and Evolution in Fourth Generation Mobile Communication Systems' *Journal of Communications and Networks* **4**(4).
[29] Decasper, D., Parulkar, G., Choi, S., DeHart, J., Wolf, T. and Plattner, B. (1999) 'A Scalable, High Performance Active Network Node', *IEEE Network*, January/February.
[30] Wetherall, David, Guttag, John and Tennenhouse, David (1999) 'ANTS: Network Services without the Red Tape' *IEEE Computer* **32**(4), 42–49.

延伸阅读

Blake, S. Black, D., Carlson, M., Davies, E., Wang, Z., Weiss, W. (1998) 'An Architecture for Differentiated Services' RFC 2475, IETF.
Braden, R. (1994) 'Integrated Services in the Internet Architecture: an Overview' RFC 1633, IETF.
Eberspächer, J., Vögel, H.-J. and Bettstetter, C. (2001) *GSM – Switching, Services, and Protocols*, 2nd edn. Wiley.
Johansson, P., Kazantzidis, M., Kapoor, R. and Gerla, M. (2001) 'Bluetooth: An Enabler for Personal Area Networking' *IEEE Network*, September.
Karl, H. and Willig, A. (2005) *Protocols and Architectures for Wireless Sensor Networks*. Wiley.
Kellerer, W., Bettstetter, C., Schwingenschlögl, C., Sties, P., Steinberg, K.-E. and Vögel, H.-J. (2001) '(Auto)mobile Communication in a Heterogeneous and Converged World' *IEEE Personal Communications Magazine*, December.
Mann, S. (1997) 'Wearable Computing: A First Step toward Personal Imaging' *IEEE Computer*, February.
Mauve, M., Hartenstein, H., Füßler, H., Widmer, J. and Effelsberg, W. (2002) 'Positionsbasiertes Routing für die Kommunikation zwischen Fahrzeugen' *it + ti*, October.
Perkins, C.E. (ed.) (2001) *Ad Hoc Networking*, Addison Wesley.
Römer, K. and Mattern, F. (2004) 'The Design Space of Sensor Networks' *IEEE Wireless Communications*, December.
Vassis, D., Kormentzas, G., Rouskas, A. and Maglogiannis, I. (2005) The IEEE 802.11g Standard for High Data Rate WLANs' *IEEE Network*, May.
Vaughan-Nichols, S.J. (2004) 'Achieving Wireless Broadband with WiMax' *IEEE Computer*, June.
Walke, B.H. (2002) *Mobile Radio Networks. Networking, Protocols and Traffic Performance*, 2nd edn. Wiley.
Walke, B.H., Althoff, M.P. and Seidenberg, P. (2001) *UMTS – Ein Kurs*. Schlembach Fachverlag.
Weiser, M. (1991) 'The Computer for the Twenty-First Century' *Scientific American*, September.
Zheng, J. and Lee, M.J. (2004) 'Will IEEE 802.15.4 Make Ubiquitous Networking a Reality?: A Discussion on a Potential Low Power, Low Bit Rate Standard' *IEEE Communications Magazine*, June.
ZigBee Alliance (2005) *ZigBee Specification*.

第 3 章　移动服务系统

参考文献

[1] ITU-T (1993) *Recommendation I.112: Vocabulary of Terms for ISDNs.*
[2] ITU-T (1993) *Recommendation Q.931: Digital Subscriber Signalling System No.1 (DSS1) – ISDN UNI Layer 3 Specification for Basic Call Control.*
[3] ITU-T (undated) *Recommendation Series Q.12xx: Intelligent Network.*
[4] Magedanz, T. and Popescu-Zeletin, R. (1996) *Intelligent Networks.* International Thomson Computer Press, London.
[5] OMG (2005) *Common Object Request Broker Architecture (CORBA/IIOP), v. 3.0.3.*
[6] Inoue, Y., Lapierre, M. and Mossotto, C. (1999) *The TINA Book – A Co-operative Solution for a Competitive World.* Prentice Hall Europe, London.
[7] Eberspächer, J., Vögel, H.-J. and Bettstetter, C. (2001) *GSM Switching, Services and Protocols,* 2nd edn. Wiley, 2001.
[8] Christensen, G., Florack, P. and Duncan, R. (2000) *Wireless Intelligent Networking,* Artech House.
[9] Holma, H. and Toskala A. (eds) (2001) *WCDMA for UMTS.* Wiley, Chichester.
[10] 3rd Generation Partnership Project (2002) *The Virtual Home Environment, version 5.1.0.*
[11] ITU-T (1998) *Recommendation H.323: Packet-Based Multimedia Communications System.*
[12] ITU-T (1998) *Recommendation H.450: Generic Functional Protocol for the Support of Supplementary Services in H.323.*
[13] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M. and Schooler, E. (2002) 'SIP – Session Initiation Protocol' RFC3261, IETF, June 2002.
[14] Glasmann, J., Kellerer, W. and Müller, H. (2003) 'Service Architectures in H.323 and SIP – A Comparison' *IEEE Communication Surveys and Tutorials,* **5**(2).
[15] Lennox, J., Wu, X. and Schulzrinne, H. (2004) 'Call Processing Language (CPL): A Language for User Control of Internet Telephony Services' RFC3880, IETF.
[16] Handley, M. and Jacobson, V. (1998) 'SDP: Session Description Protocol' RFC2327, IETF.
[17] Camarillo, G. and Garcia-Martin, M. (2004) *The 3G IP Multimedia Subsystem.* Wiley.
[18] Cuervo, F., Greene, N., Rayhan, A., Huitema, C., Rosen, B. and Segers, J. (2000) 'Megaco Protocol Version 1.0' RFC 3015, IETF.
[19] The Parlay Group (2005) 'Parlay and OSA Technical Library' [http://www.parlay.org/].
[20] Open Mobile Alliance (2005) 'Wireless Application Protocol' [http://www.openmobilealliance.org/tech/affiliates/wap/wapindex.html].
[21] Natsuno, T. (2003) *I-Mode Strategy.* Wiley.
[22] Imai, K., Takita, W., Kano, S. and Kodate, A. (2005) 'An Extension of 4G Mobile Networks towards the Ubiquitous Real Space' *IEICE Transactions on Communications,* **E88-B**(7).
[23] Arbanowski, S., Ballon, P., David, K., Droegehorn, O., Eertink, H., Kellerer, W., van Kranenburg, H., Raatikainen, K. and Popescu-Zeletin, R. (2004) 'I-centric Communications: Personalization, Ambient Awareness, and Adaptability for Future Mobile Services' *IEEE Communications Magazine* **42**(9), 63–69.
[24] Schulzrinne, H. and Wedlund, E. (2000) 'Application-Layer Mobility using SIP' *ACM MC2R* **4**(3).
[25] Raman, B., Katz, R.H. and Joseph, A.D. (2000) 'Universal Inbox: Providing Extensible Personal Mobility and Service Mobility in an Integrated Communication Network' *Workshop on Mobile Computing Systems and Applications* (WMCSA'00).
[26] Shiaa, M.M. and Aagesen, F.A. (2002) 'Architectural Considerations for Personal Mobility in the Wireless Internet' *Personal Wireless Communication* (PWC 2002), Singapore.
[27] Kikuta, Y. *et al.* (2003) 'Design of Seamless Service Environment for Adaptive Service Transfer among Terminals' *8th International Workshop on Mobile Multimedia Communications* (MoMuC 2003), Munich.
[28] Song, H., Chu, H. and Kurakake, S. (2002) 'Browser Session Preservation and Migration' *11th International World Wide Web Conference* (WWW2002), Hawaii.
[29] Bettstetter, C., Kellerer, W. and Eberspächer, J. (2000) 'Personal Profile Mobility for Ubiquitous Service Usage, Version 1.0' *Book of Visions 2000.* ISTWireless Strategic Initiative (WSI).
[30] Hasekawa, M. (2003) 'Cross-Device Handover Using the Service Mobility Proxy' *Wireless Personal Multimedia Communications (WPMC03),* Yokosuka, Japan.
[31] Shacham, R., Schulzrinne, H., Thakolsri, S. and Kellerer, W. (2005) 'The Virtual Device: Expanding Wireless

Communication Services through Service Discovery and Session Mobility' *Proceedings of IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*, WiMob'2005, Montreal, Canada, 22–24August.
[32] Shacham, R., Schulzrinne, H., Thakolsri, S. and Kellerer, W. (2007) 'Session Initiation Protocol (SIP) Session Mobility', IETF Internet Draft, November 2007, draft-shacham-sipping-session-mobility-05.txt – work in progress.
[33] Rosenberg, J., Peterson, J. Schulzrinne, H. and Camarillo G. (2004) 'Best Current Practices for Third Party Call Control (3pcc) in the Session Initiation Protocol (SIP)', RFC3725, IETF.
[34] Sparks, R. (2003) 'The Session Initiation Protocol (SIP) Refer Method', RFC 3515, IETF.
[35] Guttman, E., Perkins, C., Veizades, J. and Day, M. (1999) 'Service Location Protocol, Version 2' RFC2608, IETF.

延伸阅读

Moerdijk, L. and Klostermann, A. (2003) 'Opening the Networks with Parlay/OSA: Standards and Aspects Behind the APIs' *IEEE Network Magazine* May/June.
Roach, A. (2002) 'Session Initiation Protocol (SIP) – Specific Event Notification' RFC3265, IETF.
Rosenberg, J. (2007) 'The Extensible Markup Language (XML) Configuration Access Protocol (XCAP)' RFC4825, IETF.
Shacham, R., Schulzrinne, H., Thakolsri, S. and Kellerer, W. (2004) 'An Architecture for Location-based Service Mobility Using the SIP Event Model' *Proceedings of ACM MobiSys 2004*, Boston.
Shacham, R., Schulzrinne, H., Thakolsri, S. and Kellerer, W. (2007) 'Ubiquitous Device Personalization: The Next Generation of IP Telephony' *ACM Transactions on Multimedia Computing, Communications, and Applications* **3**(2).
Steinmetz, R. and Wehrle, K. (2005) *Peer-to-Peer-Systems and Applications* LNCS 3485. Springer.
Wagner, M. and Kellerer, W. 'Web Services Selection for Distributed Composition of Multimedia Content' *Proceedings of ACM Multimedia 2004*, New York.

第4章 泛在性扩展：移动 P2P

参考文献

[1] Sen, S. and Wang, J. (2004) 'Analyzing Peer-to-peer Traffic across Large Networks' *IEEE/ACM Transactions on Networking (TON)* **12**, 219–232.
[2] Iyer, S., Rowstron, A. and Druschel, P. (2002) 'SQUIRREL: A Decentralized, Peer-to-peer Web Cache. *Proceedings of Twenty-First ACM Symposium on Principles of Distributed Computing (PODC 2002)*, Monterey, CA.
[3] Parreira, J.X., Donato, D., Michel, S. and Weikum, G. (2006) 'Efficient and Decentralized PageRank Approximation in a Peer-to-Peer Web Search Network' *International Conference on Very Large Data Bases (VLDB)*, Seoul, Korea.
[4] Dabek, F., Kaashoek, M.F., Karger, D., Morris, R. and Stoica, I. (2001) 'Wide-area Cooperative Storage with CFS' *Proceedings of the 18th ACM Symposium on Operating Systems Principles (SOSP '01)*, Chateau Lake Louise, Banff, Canada.
[5] Stoica, I., Adkins, D., Zhuang, S., Shenker, S. and Surana, S. (2002) 'Internet Indirection Infrastructure' *Proceedings of the ACM SIGCOMM Conference.*
[6] Castro, M., Druschel, P., Kermarrec, A.-M., Nandi, A., Rowstron, A. and Singh, A. (2003) 'SplitStream: High-bandwidth Content Distribution in a Cooperative Environment. Second *International Workshop on Peer-to-Peer Systems* (IPTPS '03), Berkeley, CA.
[7] Kostic, D., Rodriguez, A., Albrecht, J. and Vahdat, A. (2003) 'Bullet: High Bandwidth Data Dissemination Using an Overlay Mesh' *Proceedings of the 19th ACM Symposium on Operating System Principles (SOSP 2003)*, New York.
[8] Caesar, M., Castro, M., Nightingale, E.B., O'Sheal, G. and Rowstron, A. (2006) 'Virtual Ring Routing: Network Routing Inspired by DHTs' In *Proceedings of the ACM SIGCOMM conference 2006*, Italy.
[9] Aberer, K., Alima, L.O., Ghodsi, A., Girdzijauskas, S., Hauswirth, M. and Haridi, S. (2005) 'The Essence of P2P: A Reference Architecture for Overlay Networks, P2P2005' *The 5th IEEE International Conference on Peer-to-Peer Computing*, Konstanz, Germany
[10] Despotovic, Z. (2005) *Building Trust-aware P2P Systems: From Trust and Reputation Management to Decen-*

tralized E-Commerce Applications. PhD Thesis (no 3313), EPFL, Switzerland.

[11] Gnutella (2001) 'Clip2. The gnutella protocol specification v0.4 (document revision 1.2)' [http://www9.limewire.com/developer/gnutella protocol 0.4.pdf].

[12] Lv, Q., Cao, P., Cohen, E., Li, K. and Shenker, S. (2002) 'Search and Replication in Unstructured Peer-to-peer Networks. *International Conference on Supercomputing*, New York.

[13] Stoica, I., Morris, R., Karger, D., Kaashoek, F. and Balakrishnan, H. (2001) 'Chord: A Scalable Peer-To-Peer Lookup Service for Internet Applications' *Proceedings of the 2001 ACM SIGCOMM Conference*, pp. 149–160.

[14] Ratnasamy, S., Francis, P., Handley, M., Karp, R. and Shenker, S. (2001) A Scalable Content-Addressable Network. *Proceedings of ACM SIGCOMM '01*, pp. 161–172.

[15] Rowstron, A. and Druschel, P. (2001) 'Pastry: Scalable, Distributed Object Location and Routing for Large-scale Peer-to-peer Systems' *IFIP/ACM International Conference on Distributed Systems Platforms (Middleware)*, pp. 329–350.

[16] Malkhi, D., Naor, M. and Ratajczak, D'. (2002) 'Viceroy: A Scalable and Dynamic Emulation of the Butterfly' *Proceedings of the 21nd ACM Symposium on Principles of Distributed Computing, PODC 2002*, pp. 183–192, Monterey, CA.

[17] Manku, G.S., Bawa, M. and Raghavan, P. (2003) 'Symphony: Distributed Hashing in a Small World' *Proceedings of the 4th USENIX Symposium on Internet Technologies and Systems (USITS 2003)*, Seattle, WA.

[18] Kleinberg, J. (2000) 'The Small-World Phenomenon: An Algorithmic Perspective' *Proceedings of the 32nd ACM Symposium on Theory of Computing (STOC 2000)*, pp. 163–170.

[19] Maymounkov, Petar and Mazières, David (2002) 'Kademlia: A Peer-to-peer Information System Based on the XOR Metric' *1st International Workshop on Peer-to-peer Systems (IPTPS2002)*, Boston, MA.

[20] Aberer, K. (2001) 'P-Grid: A Self-organizing Access Structure for P2P Information Systems' *Proceedings of the Sixth International Conference on Cooperative Information Systems (CoopIS 2001)*, Trento, Italy.

[21] Garcés-Erice, E.L., Biersack, W., Ross, K.W., Felber, P.A. and Urvoy-Keller, G. (2003) 'Hierarchical P2P Systems' *Proceedings of ACM/IFIP International Conference on Parallel and Distributed Computing (Euro-Par)*, pp. 1230–1239, Klagenfurt, Austria.

[22] Mizrak, A.T., Cheng, Y., Kumar, V. and Savage, S. (2003) 'Structured Superpeers: Leveraging Heterogeneity to Provide Constant-Time *Lookup*' *Proceedings of the Third IEEE Workshop on Internet Applications (WIAPP'03)*, pp. 104–111, San Jose, CA.

[23] Gummadi, P.K., Gummadi, R., Gribble, S.D., Ratnasamy, S., Shenker, S. and Stoica, I. (2003) 'The Impact of DHT Routing Geometry on Resilience and Proximity' In *Proceedings of the ACM SIGCOMM conference 2003*, Karlsruhe, Germany.

[24] Manku, G.S. (2003) 'Routing Networks for Distributed Hash Tables' *Proceedings of the 22nd ACM Symposium on Principles of Distributed Computing (PODC)*, pp. 133–142, Boston, MA.

[25] Steinmetz, R. and Wehrle, K. (2005) *Peer-to-Peer-Systems and Applications*, LNCS 3485. Springer.

[26] Kaashoek, M.F. and Karger, D.R. (2003) 'Koorde: A Simple Degree-optimal Distributed Hash Table' *Proceedings of the 2nd International Workshop on Peer-to-Peer Systems (IPTPS '03)*, Berkeley, CA.

[27] Rhea, S., Geels, D., Roscoe, T. and Kubiatowicz, J. (2004) 'Handling Churn in a DHT' *Proceedings of the 2004 USENIX Annual Technical Conference, (USENIX '04)*, Boston, MA.

[28] Harren, M., Hellerstein, J.M., Huebsch, R., Loo, B.T., Shenker, S. and Stoica, I. (2002) 'Complex Queries in DHT-based Peer-to-Peer Networks' *First International Workshop on Peer-to-Peer Systems (IPTPS '02)*, Cambridge, MA.

[29] Datta, A., Hauswirth, M., John, R., Schmidt, R. and Aberer, K. (2005) 'Range Queries in Trie-structured Overlays' *Proceedings of the 5th IEEE conference on P2P Computing*, Konstanz, Germany.

[30] Triantafillou, P. and Pitoura, T. (2003) 'Towards a Unifying Framework for Complex Query Processing over Structured Peer-to-Peer Data Networks' *VLDB '03 Workshop on Databases, Information Systems, and Peer-to-Peer Computing*, Germany.

[31] Ramabhadran, S., Ratnasamy, S., Hellerstein, J.M. and Shenker, S. (2004) 'Brief Announcement: Prefix Hash Tree' *Proceedings of ACM PODC*, St Johns, Canada.

[32] Klemm, F., Datta, A. and Aberer, K. (2004) 'A Query-Adaptive Partial Distributed Hash Table for Peer-to-Peer Systems' *International Workshop on Peer-to-Peer Computing & DataBases (P2P&DB 2004)*, Crete.

[33] Garcés-Erice, L., Felber, P., Biersack, E.W., Urvoy-Keller, G. and Ross, K.W. (2004) 'Data Indexing in Peer-to-Peer DHT Networks' *Proceedings of the 24th International Conference on Distributed Computing Systems*

(ICDCS 2004), pp. 200–208, Tokyo.
[34] Skobeltsyn, G., Hauswirth, M. and Aberer, K. (2005) 'Efficient Processing of XPath Queries with Structured Overlay Networks' *Proceedings of ODBASE'05*, Agia Napa, Cyprus.
[35] Lakshminarayanan, Karthik, Rao, Ananth, Stoica, Ion and Shenker, Scott (2005) 'End-host Controlled Multicast Routing', *Elsevier Computer Networks*, Special Issue on Overlay Distribution Structures and their Applications.
[36] Castro, M., Druschel, P., Kermarrec, A.-M. and Rowstron, A. (2003) 'Scalable Application-level Anycast for Highly Dynamic Groups' *NGC 2003*, Munich, Germany.
[37] Aekaterinidis, I. and Triantafillou, P. (2006) 'PastryStrings: A Comprehensive Content-Based Publish/Subscribe DHT Network' *26th IEEE International Conference on Distributed Computing and Systems (ICDCS 06)*, Portugal.
[38] Kellerer, W., Schollmeier, R. and Wehrle, K. (2005) 'Peer-to-peer in Mobile Environments. In Steinmetz, R. and Wehrle, K. (eds), LNCS Volume 3485. Springer.
[39] Kirk, P. (2003) 'The Gnutella Protocol Specification' [http://rfc-gnutella.sourceforge.net, v. 0.6].
[40] Zöls, S., Schollmeier, R., Kellerer, W. and Tarlano, A. (2005) 'The Hybrid Chord Protocol: A Peer-to-peer Lookup Service for Context-aware Mobile Applications' *Proceedings of 2005 International Conference on Networking (ICN'05)*, Réunion Island, France.
[41] Zöls, S., Schubert, S., Despotovic, Z. and Kellerer, W. (2006) 'Hybrid DHT Design for Mobile Environments' *5th International Workshop on Agents and P2P Computing (AP2PC 06)*, held at AAMAS 06, Japan.
[42] Zöls, S., Despotovic, Z. and Kellerer, W. (2005) 'Cost-based Analysis of Hierarchical DHT Design' *Proceedings of the 6th IEEE conference on P2P Computing*, Cambridge, UK.
[43] Gruber, I., Schollmeier, R. and Kellerer, W.' (2004) Performance Evaluation of the Mobile Peer-to-peer Protocol' *Proceedings of ACM/IEEE International Workshop on Global Peer-to-Peer Computing*, 19–22 April, Chicago, IL.
[44] Gruber, I., Schollmeier, R., Kellerer, W. (2006) 'Peer-to-peer Communication in Mobile Ad-Hoc Networks'. *Ad Hoc & Sensor Wireless Networks. An International Journal* **2**(2.3).
[45] Johnson, D. and Maltz, D. (2001) 'Dynamic Source Routing in ad hoc Wireless Networks. In Perkins, C.E. (ed.), *Ad Hoc Networking*, pp. 139–172. Addison-Wesley.
[46] Resnick, P., Zeckhauser, R., Friedman, E. and Kuwabara, K. (2000) 'Reputation Systems' *Communications of the ACM*, **43**(12), 45–48.
[47] Kamvar, S.D., Schlosser, M.T. and Garcia-Molina, H. (2003) 'EigenRep: Reputation Management in P2P Networks' *Proceedings of the World Wide Web Conference*, Budapest.
[48] Despotovic, Z. and Aberer, K. (2004) 'A Probabilistic Approach to Predict Peers' Performance in P2P Networks' *Eighth International Workshop on Cooperative Information Agents*, Erfurt, Germany.
[49] Kreps, D. and Wilson, R. (1982) 'Reputation and Imperfect Information' *Journal of Economic Theory*, **27**, 253–279.
[50] Blanc, A., Liu, Y.-K. and Vahdat, A. (2005) 'Designing Incentives for Peer-to-peer Routing' *Proceedings of the IEEE Infocom Conference*, Miami, FL.

第5章 移动中间件

参考文献

[1] Mattern, F. and Sturm, P. (2003) 'From Distributed Systems to Ubiquitous Computing' *Fachtagung Kommunikation in verteilten Systemen (KIVS)*, February, pp. 3–25.
[2] Bakken, D.E. (2003) 'Middleware' In Dasgupta, J. (ed.) *Encyclopaedia of Distributed Computing*, Kluwer.
[3] Gaddah, A. and Kunz, T. (2003) *A Survey of Middleware Paradigms for Mobile Computing*, Technical report sce-03-16, Carleton University, Ottawa.
[4] OMG (2004) 'CORBA Notification Service, Version 1.1' [http://www.omg.org].
[5] Sun Microsystems (2002) 'Java Message Service (JMS) version 1.1' [http://java.sun.com/products/jms].
[6] Cugola, G. and Di Nitto, E. (2001) 'Using a Publish/Subscribe Middleware to Support Mobile Computing' *Advanced Topic Workshop on Middleware for Mobile Computing*, Hiderberg, Germany.
[7] Smith, B. (1982) *Reflection and Semantics in a Procedural Programming Languages*. Report, MIT, Cambridge, MA.
[8] Schmidt, D.C. and Cleeland, C. (1999) 'Applying Patterns to Develop Extensible and Maintainable ORB

Middleware' *IEEE Communication Magazine*, Special Issue on Design Patterns, April.
[9] Kon, F., Gill, B., Anand, M., Campbell, R. and Dennis Mickunas, M. (2000) 'Secure Dynamic Reconfiguration of Scalable CORBA Systems with Mobile Agents' *IEEE Joint Symposium on Agent Systems and Applications/Mobile Agents*, Zurich.
[10] Mascolo, C., Capra, L., Zachariadis, S. and Emmerich, W. (2002) 'XMIDDLE: A Data-Sharing Middleware for Mobile Computing' *Personal and Wireless Communications Journal* **21**(1).
[11] Capra, L., Mascolo, C., Zachariadis, S. and Emmerich, W. (2001) 'Towards a Mobile Computing Middleware: a Synergy of Reflection and Mobile Code Techniques' *8th IEEE Workshop on Future Trends of Distributed Computing Systems (FTDCS'2001)*, Bologna, Italy. IEEE Computer Society Press.
[12] Karjoth, G., Lange, D. and Oshima, M. (1997) 'A security model for Aglets' *IEEE Internet Computing Magazine*, July/August.
[13] Bellavista, P., Corradi, A. and Stefanelli, C. (2001) 'Mobile Agent Middleware for Mobile Computing' *IEEE Computer Magazine* **34**(3), 73–81.
[14] emporphia Ltd (undated) 'FIPA-OS' [http://www.emorphia.com/research/about.htm].
[15] Telecom Italia Lab. (undated) 'JADE' [http://jade.cselt.it].
[16] EU IST Project (undated) 'Lightweight Extensible Agent Platform (LEAP)' [http://leap.crm-paris.com].
[17] Project JXTA (undated) 'JXTA' [http://www.jxta.org].
[18] Steglich, S., Vaidya, R. N., Gimpeliovskaja, O., Arbanouski, S., Popesku-Zeletin, R., Sameshima, S., Kawano, K. (2003) 'I-Centric Services Based on Super Distributed Objects' *ISADS 2003 The Sixth International Symposium*.
[19] Apple Computer (2005) 'Bonjour' [http://www.apple.com/macosx/features/bonjour].
[20] IETF (2005) 'Dynamic Configuration of IPv4 Link-Local Addresses' RFC3927.
[21] Fenkam, P., Kinda, E., Dustar, S., Gall, H., Reif, G. (2002) 'Evaluation of a Publish/Subscribe System for Collaborative and Mobile Working' *The 11th International Workshop on Enabling Technologies, Infrastructure for Collaborative Enterprise* (WETICE'02), Pittsburgh, PA.
[22] Kon, F., Gill, B., Anand, M., Campbell, R. and Mickunas, M. Dennis (2000) 'Secure Dynamic Reconfiguration of Scalable CORBA Systems with Mobile Agents' *IEEE Joint Symposium on Agent Systems and Applications/Mobile Agents*, Zurich.
[23] The 3rd Generation Partnership Project (3GPP) (2005) [http://www.3gpp.org/.
[24] European Telecommunications Standards Institute Smart Card Platform (ETSI SCP) (undated) [http://portal.etsi.org/scp/summary.asp.
[25] EMV 2000 version 4.1 (undated) [http://www.emvco.com.
[26] Sun Microsystems (2003) 'Java Card version 2.2.1 [http://java.sun.com/products/javacard].
[27] MULTOS version 5 (2000) [http://www.multos.com.
[28] Octpus (2005) [http://www.octopuscards.com//enindex.jsp.
[29] Oyster (2005) [http://www.oystercard.com.
[30] RATP (2005) 'Navigo Pass' [http://www.ratp.fr/corpo/service/navigo.html].
[31] NTT DoCoMo (undated) 'i-mode FeliCa' [http://www.nttdocomo.com/corebiz/services/imode/felica.html]
[32] Sony (undated) 'FeliCa' [http://www.sony.net/Products/felica].
[33] Sun Microsystems (2005) 'Connected Limited Device Configuration' [http://java.sun.com/products/cldc].
[34] Finkenzeller, K. (1999) *RFID-Handbook*. Wiley.
[35] Nokia (undated) 'RFID in Brief' [http://www.nokia.com/nokia/0,,55738,00.html].
[36] Solarski, M., Strick, L., Motonaga, K., Noda, C. and Kellerer, W. (2004) 'Flexible Middleware Support for Future Mobile Services and Their Context-Aware Adaptation' *Proceedings of Lecture Notes in Computer Science*, pp. 281–292. Springer.
[37] Noda, C. and Walter, T. (2004) 'Smart Devices for Next Generation Mobile Services' *Proceedings of CASSIS*, Marseille, pp. 192–209. Springer.
[38] Blefari-Melazzi, N., Kellerer, W., Noda, C., Salsano, S. and Seidl, R. (2005) 'The Simplicity Project: Architecture Concept' IEEE/IPSJ SAINT2005. *The 2005 International Symposium on Applications and the Internet, workshop on 'Next Generation IP-based Service Platforms for Future Mobile Systems'*, Trento, Italy, 31 January–4 February.

延伸阅读

Blair, G.S., Coulson, G., Andersen, A., Blair, L., Clarke, M., Costa, F., Duran-Limon, H., Fitzpatrick, T., Johnston, L., Moreira, R., Parlavantzas, N., Saikoski, K. (2001) 'The Design and Implementation of Open ORB v2' *IEEE*

DS Online, Special Issue on Reflective Middleware.
Bologna University (undated) 'SOMA' [http://www-lia.deis.unibo.it/Research/SOMA].
International Organization for Standards (undated) [http://www.iso.org.
Padovitz, A., Loke, S.W., and Zaslavsky, A. (2003) 'Using the Publish/Subscribe Genre for Mobile Agents' Proceedings *of 1st German Conf. on Multiagent System Technology (MATES'03)*, Erfurt, Germany, September, pp. 180–192. Springer-Verlag.

第6章 跨层设计—— 一种新的移动通信系统优化方法

参考文献

[1] Zimmermann, H. (1980) 'OSI Reference Model – The ISO Model of Architecture for Open Systems Interconnection' *IEEE Transactions on Communications* **28**(4), 425–432.
[2] Kawadia, V. and Kumar, P. (2003) 'A Cautionary Perspective on Cross Layer Design' *IEEE Wireless Communication Magazine*, July.
[3] Shakkottai, S., Rappaport, T. and Karlsson, P. (2003) 'Cross-layer Design for Wireless Networks' *IEEE Communications Magazine*, October.
[4] Clark, D. and Tennenhouse, D. (1990) 'Architectural Considerations for a New Generation of Protocols *Computer Communication Review*, ACM SIGCOMM '90 Symposium.
[5] Haas, Z. (2001) 'Design Methodologies for Adaptive and Multimedia Networks' *IEEE Communication Magazine*, November.
[6] Rappaport, T., Annamalai, A., Buehrer, R. and Tranter, W. (2002) 'Wireless Communications: Past Events and a Future Perspective' *IEEE Communications Magazine* **40**(5), 148–161.
[7] Adve, S., Harris, A., Hughes, C., Jones, D., Kravets, R., Nahrstedt, K., Sachs, D., Sasanka, R., Srinivasan, J. and Yuan, W. (2002) 'The Illinois GRACE Project: Global Resource Adaptation through Cooperation *Proceedings of the Workshop on Self-Healing, Adaptive, and Self-managed Systems* (held in conjunction with the 16th Annual ACM International Conference on Supercomputing), June.
[8] Ludwig, R. (1999) *A Case for Flow-adaptive Wireless Links* Technical Report UCB//CSD-99-1053. University of California at Berkeley.
[9] Sternad, M. (2002) *The Wireless IP Project*. RVK, Stockholm.
[10] Chen, K., Shah, H. and Nahrstedt, K. (2002) 'Cross-Layer Design for Data Accessibility in Mobile Ad-hoc Networks' *Wireless Personal Communications* **21**(1).
[11] Yuen, W., Lee, H. and Andersen, T. (2002) 'A Simple and Effective Cross Layer Networking System for Mobile ad hoc Networks' *Proceedings of IEEE PIMRC.*
[12] Peng, Y., Khan, S., Steinbach, E., Sgroi, M. and Kellerer, W. (2005) 'Adaptive Resource Allocation and Frame Scheduling for Wireless Multi-User Video Streaming' *Proceedings of ICIP 2005.*
[13] Gross, J., Klaue, J., Karl, H. and Wolisz, A. (2004) 'Cross-Layer Optimization of OFDM Transmission Systems for MPEG-4 Video Streaming' *Computer Communications*, **27**, 1044–1055.
[14] Zhang, Q., Zhu, W. and Zhang, Y. (2002) A Cross-layer QoS-supporting Framework for Multimedia Delivery over Wireless Internet' *International Packet Video Workshop 2002.*
[15] Tupelly, R.S., Zhang, J. and Chong, E.K.P. (2003) 'Opportunistic Scheduling for Streaming Video in Wireless Networks' *Proceedings of the 37th Annual Conference on Information Sciences and Systems*, Baltimore, MD, 12–14 March.
[16] Xylomenos, G. and Polyzos, G. (1999) 'Link Layer support for Quality of Service on Wireless Internet Links' *IEEE Personal Communication Magazine* **6**(5), 52–60.
[17] Zhang, Y. and Cheng, L. (2003) 'Cross-Layer Optimization for Sensor Networks' *New York Metro Area Networking Workshop 2003*, New York City, 12 September.
[18] Khan, S., Sgroi, M., Peng, Y., Steinbach, E. and Kellerer, W. (2006) 'Application-driven Cross Layer Optimization for Video Streaming over Wireless Networks' *IEEE Communications Magazine*, Special Issue on Cross-Layer Protocol Engineering **44**(1).
[19] Krunz, M. and Tripathi, S.K. (1997) 'Exploiting the Temporal Structure of MPEG Video for the Reduction of Bandwidth Requirements' *Proceedings of INFOCOM 1997.*
[20] Tu, W., Kellerer, W. and Steinbach, E. (2004) 'Rate-Distortion Optimized Video Frame Dropping on Active Network Nodes' *Packet Video Workshop 2004*, Irvine, CA, 13–14 December.
[21] Khan, S., Sgroi, M., Steinbach, E. and Kellerer, W. (2005) 'Cross-Layer Optimization for Wireless Video Streaming – Performance and Cost' *Proceedings of ICME 2005.*

[22] Khan, S., Duhovnikov, S., Steinbach, E., Sgroi, M. and Kellerer, W. (2006) 'Application-driven Cross-layer Optimization for Mobile Multimedia Communication using a Common Application Layer Quality Metric' *Proceedings of 2nd IEEE International Symposium on Multimedia over Wireless (ISMW 2006)*, as part of International Wireless Communications and Mobile Computing Conference (IWCMC 2006), Vancouver 3–6 July 3–6.

第 7 章　本体

参考文献

[1] Gruber, Th. (1993) 'A Translation Approach to Portable Ontology Specifications' *Knowledge Acquisition* **5**(2), 199–220.

[2] Staab, S. and Studer, R. (eds) (2004) *Handbook on Ontologies*. Springer.

[3] Baader, F., Calvanese, D., McGuinness, D., Nardi, D. and Patel-Schneider, P. (2003) *The Description Logic Handbook – Theory, Implementation and Applications*. Cambridge University Press.

[4] Schmidt-Schauß, M. and Smolka, G. 'Attributive Concept Descriptions with Complements' *Artificial Intelligence* **48**(1), 1–26.

[5] Sattler, U. (1996) 'A Concept Language Extended with Different Kinds of Transitive Roles' In Görz, G. and Hölldobler, S. (eds), *20. Deutsche Jahrestagung für Künstliche Intelligenz*, number 1137 in *Lecture Notes in Artificial Intelligence*. Springer.

[6] Baader, F. and Hanschke, Ph. (1991) 'A Schema for Integrating Concrete Domains into Concept Languages' *Proceedings of the 12th International Conference on Artificial Intelligence (IJCAI'91)*, pp. 452–457.

[7] Horrocks, I. (1997) *Optimising Tableaux Decision Procedures for Description Logics*. PhD thesis, University of Manchester.

[8] Horrocks, I. and Sattler, U. (1999) 'A Description Logic with Transitive and Inverse Roles and Role Hierarchies' *Journal of Logic and Computation* **9**(3), 385–410.

[9] Horrocks, I., Sattler, U. and Tobies, S. 'Practical Reasoning for Very Expressive Description Logics' *Logic Journal of the IGPL* **8**(3), 239–263.

[10] Horrocks, I. and Sattler, U. (2001) 'Ontology Reasoning in the SHOQ(D) Description Logic' In Nebel, B. (ed.), *Proceedings of the 17th International Joint Conference on Artificial Intelligence (IJCAI'01)*, pp. 199–204. Morgan Kaufmann.

[11] Horrocks, I., Li, L., Turi, D. and Bechhofer, S. (2004) The Instance Store: DL reasoning with large numbers of individuals. In Haarslev, V. and Möller, R. (eds), *International Workshop on Description Logics (DL'04)*, pp. 31–40. Whistler, British Columbia, Canada, June.

[12] Lutz, C., Areces, C., Horrocks, I. and Sattler, U. (2004) 'Keys, Nominals, and Concrete Domains *Journal of CEUR Workshop Proceedings* **49**, 170–179.

[13] Haarslev, V., Möller, R. and Wessel, M. (2005) 'Description Logic Inference Technology: Lessons Learned in the Trenches' In Horrocks, I. *et al.* (eds.), *Proceedings of the International Workshop on Description Logics (DL'05)*, pp. 160–167, July.

[14] Tobies, S. (2000) 'The Complexity of Reasoning with Cardinality Restrictions and Nominals in Expressive Description Logics' *Journal of Artificial Intelligence Research* **12**, 199–217.

[15] Horrocks, I. and Sattler, U. (2005) A Tableaux Decision Procedure for SHOIQ' *Proceedings of 19th International Joint Conference on Artificial Intelligence (IJCAI'05)*.

[16] Brachman, R. and Schmolze, J. (1985) 'An Overview of the KL-ONE Knowledge Representation System' *Cognitive Science* **9**(2), 171–216.

[17] MacGregor, R. (1991) 'Using a Description Classifier to Enhance Deductive Inference' *Proceedings of the 7th IEEE Conference on AI Applications*, pp. 141–147.

[18] Baader, F., Hollunder, B., Nebel, B., Profitlich, H.-J. and Franconi, E. (1994) 'An Empirical Analysis of Optimization Techniques for Terminological Representation Systems' *Applied Intelligence* **4**(2), 109–132

[19] Horrocks, I. (1998) 'Using an Expressive Description Logic: Fact or Fiction?' In Cohn, A.G., Schubert, L. and Shapiro, S.C. (eds), *Proceedings of the Sixth International Conference on Principles of Knowledge Representation and Reasoning (KR'98)*, pp. 636–647, San Francisco, CA, June. Morgan Kaufmann.

[20] Haarslev, V. and Möller, R. (2001) 'Racer System Description' *International Joint Conference on Automated Reasoning (IJCAR'01)* 18–23 June, Siena, Italy. Springer.

[21] Sirin, E. and Parsia, B. (2004) 'Pellet: An OWL DL Reasoner' In Haarslev, V. and Möller, R. (eds), *International*

Workshop on Description Logics (DL'04), pp. 212–213. Whistler, British Columbia, Canada, June.
[22] Tsarkov, D. and Horrocks, I. (2004) 'Efficient Reasoning with Range and Domain Constraints. In Haarslev, V. and Möller, R. (eds), *International Workshop on Description Logics (DL'04)*, pp. 41–50. Whistler, British Columbia, Canada, June.
[23] Horrocks, I. (2005) 'Applications of Description Logics: State of the Art and Research Challenges' *Proceeedings of the 13th Inerntional Conference on Conceptual Structures (ICCS'05)*.
[24] Baader, F., Horrocks, I. and Sattler, U. (2003) 'Description Logics as Ontology Language for the Semantic Web' In Hutter, D. and Stephan, W. (eds), *Festschrift in Honor of Jörg H. Siekmann*. Springer.
[25] Berners-Lee, T., Hendler, J. and Lassila, O. (2001) 'The Semantic Web' *Scientific American* **284**(5), 34–43.
[26] Bechhofer, S., van Harmelen, F., Hendler, J., Horrocks, I., McGuinness, D., Patel-Schneider, P.F. and Andrea Stein, L. (2004) *OWL Web Ontology Language Reference* W3C Recommendation. The Worldwide Web Consortium.
[27] Horrocks, I. (2002) 'DAML+OIL: a Description Logic for the Semantic Web' *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering* **25**(1), 4–9.
[28] Horrocks, I., Patel-Schneider, P. and van Harmelen, F. (2003) 'From SHIQ and RDF to OWL: The Making of a Web Ontology Language' *Journal of Web Semantics*, **1**(1).
[29] Patel-Schneider, P.F., Hayes, P. and Horrocks, I. (2004) *OWL Web Ontology Language Semantics and Abstract Syntax* W3C recommendation. The Worldwide Web Consortium.
[30] McGuinness, D. and van Harmelen, F. (2004) OWL *Web Ontology Language Overview* W3C Recommendation. The Worldwide Web Consortium.
[31] Tessaris, S. (2001) 'Querying Expressive DLs' *Proceedings of the 2001 International Description Logics Workshop (DL'01)*.
[32] Horrocks, I. and Tessaris, S. (2002) 'Querying the Semantic Web: a Formal Approach. In Horrocks, Ian and Hendler, James (eds), *Proceedings of the 2002 International Semantic Web Conference (ISWC'02)*, Volume 2342 of *Lecture Notes in Computer Science*, pp. 177–191. Springer.
[33] Fikes, R., Hayes, P. and Horrocks, I. (2004) 'OWL-QL – A Language for Deductive Query Answering on the Semantic Web *Journal of Web Semantics* **2**(1), 19–29.
[34] Wessel, M. and Möller, R. (2005) A high performance semantic web query answering engine. In Horrocks, I., Sattler, U. and Wolter, F. (eds) *International Workshop on Description Logics (DL'05)*, Edinburgh, Scotland, July, pp. 584–595. National e-Science Centre.
[35] Karvounarakis, G., Alexaki, S., Christophides, V., Plexousakis, D. and Scholl, M. (2003) 'Querying the Semantic Web with RQL' *Computer Networks* **42**(5), 617–640.
[36] Seaborne, A. (2004) *RDQL – a Query Language for RDF* W3C member submission. The Worldwide Web Consortium.
[37] Sintek, M. and Decker, St. (2002) 'TRIPLE – a Query, Inference, and Transformation Language for the Semantic Web' *International Semantic Web Conference (ISWC)*, Sardinia, June.
[38] Haase, P., Broekstra, J., Eberhart, A. and Volz, R. (2004) 'A Comparison of RDF Query Languages *Proceedings of the Third International Semantic Web Conference (ISWC'04)*, Hiroshima, Japan.
[39] Preece, A. (1996) 'Validating Dynamic Properties of Rule-Based Systems' *Journal of Human-Computer Studies* **44**, 145–169.
[40] Winston, P. (1994) *Artificial Intelligence*. Addison-Wesley.
[41] McBride, B. (2001) 'Jena: Implementing the RDF Model and Syntax Specification' *Proceedings of the 2nd International Workshop on the Semantic Web (SemWeb'01)*, Hongkong, May.
[42] Bruijn, J. de and Fensel, D. (2005) *OWL-. WSML Deliverable* WSML Working Draft D20.1 v0.2, 6 January. Digital Enterprise Research Institute (DERI).
[43] Schmidt-Schauß, M. (1989) 'Subsumption in KL-ONE is Undecidable' *Proceedings of the 1st Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR'89)*, pp. 421–431. Morgan Kaufmann.
[44] Motik, B., Sattler, U. and Struder, R. (2004) 'Query Answering for OWL DL with Rules' *Proceedings of the Third International Semantic Web Conference (ISWC'04)*, Hiroshima, Japan, 2004. Defines the decidable DL-safe fragment of the Semantic Web Rule Language (SWRL).
[45] Grosof, B.N., Horrocks, I., Volz, R. and Decker, St. (2003) 'Description Logic Programs: Combining logic programs with Description Logic' *Procedings of the Twelfth International World Wide Web Conference (WWW 2003)*, pages 48–57. ACM, 2003.
[46] Levy, A. and Rousset, M.-Ch. (1998) 'Combining Horn Rules and Description Logics in CARIN' *Artificial Intelligence* **104**(1–2), 165–209.

[47] Horrocks, I. and Patel-Schneider, P. (2004) 'A Proposal for an OWL Rules Language' *Proceedings of the Thirteenth International World Wide Web Conference (WWW'04)*, pp. 723–731. ACM.

[48] Grau, B., Parsia, B. and Sirin, E. (2004) 'Working with Multiple Ontologies on the Semantic Web' *Proceedings of the Third International Semantic Web Conference (ISWC'04)*, Hiroshima, Japan. Springer.

[49] Kalyanpur, A., Parsia, B. and Hendler, J. (2005) 'A Tool for Working with Web Ontologies' *International Journal on Semantic Web and Information Systems* **1**(1), 36–49.

[50] Heflin, J. (2004) *OWL Web Ontology Language Use Cases and Requirements* W3C Recommendation. The Worldwide Web Consortium [http://www.w3.org/TR/webont-req].

[51] Heflin, J. and Muñoz-Avila, H. (2004) *Integrating HTN Planning and Semantic Web Ontologies for Efficient Information Integration* Technical Report LU-CSE-04-002. Department of Computer Science and Engineering, Lehigh University.

[52] Liebig, Th., Pfeifer, H. and von Henke, F. (2004) 'Reasoning Services for an OWL Authoring Tool: An Experience Report' In Haarslev, V. and Möller, R. (eds), *International Workshop on Description Logics (DL2004)*, pp. 79–82, Whistler, British Columbia, Canada, June.

[53] Bechhofer, S. (2003) 'The DIG Description Logic Interface: DIG/1.1' *Proceedings of the 2003 International Workshop on Description Logics (DL'03)*, Rome, Italy, June.

[54] Chen, C., Haarslev, V. and Wang, J. (2005) 'LAS: Extending Racer by a large abox store' In Horrocks, I., Sattler, U. and Wolter, F. (eds) *International Workshop on Description Logics (DL'05)*, Edinburgh, Scotland, July, pp. 200–207. National e-Science Centre.

[55] Gruber, Th. (1995) 'Towards Principles for the Design of Ontologies used for Knowledge Sharing' *International Journal of Human-Computer Studies* **43**, 907–928.

[56] Zhdanova, A.V. and Keller, U. (2005) 'Choosing an Ontology Language' *Proceedings of the Second World Enformatika Congress (WEC'05)*, pp. 47–50, Istanbul, Turkey, February.

[57] Martin, Ph. (2000) 'Conventions and Notations for Knowledge Representation and Retrieval' In B. Ganter and G.W. Mineau (eds), *Proceedings of the 8th International Conference on Conceptual Structures (ICCS'00)*, LNAI, vol. 1867, pp. 41–54, August. Springer-Verlag.

[58] Guha, R.V. and Bray, T. (1997) *Meta Content Framework using XML* W3C Technical Report. World Wide Web Consortium.

[59] Masolo, C., Borgo, S., Gangemi, A., Guarino, N. and Oltramari, A. (2003) *Ontology Library* WonderWeb Deliverable D18, EU-Project IST-2001-33052.

[60] Akkermans, H., Brown, M., Bouladoux, J.-M., Dieng, R., Ding, Y., Gómez-Pérez, A., *et al.* (2002) *Successful Scenarios for Ontology-based Applications* EU-Project IST-2000-29243 OntoWeb, Deliverable D21.

[61] Noy, N.F., Sintek, M., Decker, S., Crubezy, M., Fergerson, R.W. and Musen, M.A. (2001) 'Creating Semantic Web contents with Protégé-2000' *IEEE Intelligent Systems* **16**(2), 60–71.

[62] Gangemi, A., Guarino, N., Masolo, C., Oltramari, A. and Schneider, L. (2002) 'Sweetening Ontologies with DOLCE' In A. Gomez-Perez and V.R. Benjaminis (eds), *Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management. Ontologies and the Semantic Web*, Vol. 2473, pp. 166–181. Springer.

[63] Bateman, J. and Farrar, S. (2004) 'Towards a Generic Foundation for Spatial Ontology' In *Proceedings of the 4th International Conference on Formal Ontology in Information Systems (FOIS'04)*, Torino, Italy.

[64] Tonti, G., Bradshaw, J.M., Jeffers, R., Montanari, R., Suri, N., Uszok, A. (2003) 'Semantic Web Languages for Policy Representation and Reasoning: A Comparison of KAoS, Rei, and Ponder' In Fensel, D., Sycara, K.P. and Mylopoulos, J. (eds), *Proceedings of the 2nd International Semantic Web Conference (ISWC'03)*, LNCS, October. Springer.

[65] Ding, L., Zhou, L., Finin, T. and Joshi, A. (2005) 'How the Semantic Web is being Used: An Analysis of FOAF' *Proceedings of the 38th International Conference on System Sciences*, January.

[66] Chen, H., Perich, F., Finin, T. and Joshi, A. (2004) 'SOUPA: Standard Ontology for Ubiquitous and Pervasive Applications' *International Conference on Mobile and Ubiquitous Systems: Networking and Services*, pp. 258–267, Boston, MA, August.

[67] Hobbs, J.R. and Pan, F. (2004) 'An Ontology of Time for the Semantic Web' *ACM Transactions on Asian Language Information Processing* **3**(1), 66–85.

[68] Niles, I. and Pease, A. (2001) 'Towards a Standard Upper Ontology' In Welty, Chris and Smith, Barry (eds), *Proceedings of the 2nd International Conference on Formal Ontology in Information Systems (FOIS'01)*, Ogunquit, ME.

[69] Chen, H., Finin, T. and Joshi, A. (2004) 'An Ontology for Context-aware Pervasive Computing Environments *Knowledge Engineering Review*, Special Issue on Ontologies for Distributed Systems **18**(3), 197–207.

[70] Wang, X., Zhang, D., Dong, J., Chin, Ch. and Hettiarachchi, S. (2004) 'Semantic Space: A Semantic Web Infrastructure for Smart Spaces' *IEEE Pervasive Computing*, **3**(3), 32–39.

[71] Martin, D., Burstein, M., Hobbs, J., Lassila, O., McDermott, D., McIlraith, S., Narayanan, S., Paolucci, M., Parsia, B., Payne, T., Sirin, E., Srinivasan, N. and Sycara, K. (2004) *OWL-S: Semantic Markup for Web Services* W3C Member Submission.

[72] Semy, S., Pulvermacher, M. and Obrst, L. (2004) *Towards the Use of an Upper Ontology for U.S. Government and Military Domains: An Evaluation.* Technical Report TR-04-0603, MITRE, Bedford, MA.

[73] Farrar, S. and Bateman, J. (2004) *General Ontology Baseline*. OntoSpace German Project on Spatial Cognition, SFB/TR8. Deliverable D1, University of Bremen.

[74] Horrocks, I. and Voronkov, A. (2006) 'Reasoning Support for Expressive Ontology Languages Using a Theorem Prover' *Proceedings of the Fourth International Symposium on Foundations of Information and Knowledge Systems (FoIKS)*, LNCS 3861, pp. 201–218, Springer.

[75] Aftelak, A., Häyrynen, A., Klemettinen, M. and Steglich, S. (2004) 'MobiLife: Applications and Services for the User-centric Wireless World' *IST Mobile and Wireless Communications Summit*, Lyon, France, June.

[76] Floréen, P., Przybilski, M., Nurmi, P., Koolwaaij, J., Tarlano, A., Wagner, M., Luther, M., Bataille, F., Boussard, M., Mrohs, B. and Lau, S. (2005) 'Towards a Context Management Framework for MobiLife' *Proceedings of the 14th IST Mobile and Wireless Communication Summit (MOWICON'05)*, Dresden, Germany, 29–23 June 2005.

[77] Dawson, F. and Howes, T. (1998) *vCard*. The Internet Society.

[78] Brickley, D. and Miller, Li. (2005) *FOAF Vocabulary Specification.* Namespace Document.

[79] Mika, P. and Gangemi, A. (2004) 'Descriptions of Social Relations' *Proceedings of the 1st Workshop on Friend of a Friend, Social Networking and the Semantic Web*, Galway, Ireland, September.

[80] Pan, F. and Hobbs, J. (2004) 'Time in OWL-S' *Proceedings of the AAAI Symposium.*

[81] Allen, J.F. (1983) 'Maintaining Knowledge about Temporal Intervals' *Communications of the ACM*, **26**(11), 832–843.

[82] Randell, D.A., Cui, Z. and Cohn, Anthony G. (1992) 'A Spatial Logic based on Regions and Connections' *Proceedings of the Third International (KR'92)*, pp. 165–176. Morgan Kaufman.

[83] Liebig, Th. and Noppens, O. (2004) 'OntoTrack: Combining Browsing and Editing with Reasoning and Explaining for OWL Lite Ontologies' *Proceedings of the 3rd International Semantic Web Conference (ISWC'04)*, Hiroshima, Japan, November.

[84] Knublauch, H., Fergerson, R.W., Noy, N.F. and Musen, M.A. (2004) 'The Protégé OWL Plugin: An Open Development Environment for Semantic Web Applications' *Proceedings of the Third International Semantic Web Conference (ISWC'04)*, Hiroshima, Japan.

[85] Grau, B., Parsia, B., Sirin, E. and Kalyanpur, A. (2005) 'Automatic Partitioning of OWL Ontologies using $\mathcal{E}$-connections' In Horrocks, I., Sattler, U. and Wolter, F. (eds) *International Workshop on Description Logics (DL'05)*, Edinburgh, Scotland, July. National e-Science Centre.

[86] Grau, B., Parsia, B. and Sirin, E. (2006) 'Combining OWL Ontologies using $\mathcal{E}$-connections' *Journal of Web Semantics* **4**(1), 40–59.

[87] Kalyanpur, A., Parsia, B. and Sirin, E. Black box techniques for debugging unsatisfiable concepts' In Horrocks, I., Sattler, U. and Wolter, F. (eds) *International Workshop on Description Logics (DL'05)*, Edinburgh, Scotland, July. National e-Science Centre.

[88] Parsia, B., Sirin, E. and Kalyanpur, A. (2005) 'Debugging OWL Ontologies' *Proceedings of the 14th International World Wide Web Conference (WWW'05)*, May.

[89] Liebig, Th. and Halfmann, M. (2005) 'Explaining Subsumption in $\mathcal{ALEHFR}$+ Tboxes. In Horrocks, I., Sattler, U. and Wolter, F. (eds) *International Workshop on Description Logics (DL'05)*, Edinburgh, Scotland, July. National e-Science Centre.

延伸阅读

Luther, M., Mrohs, B., Wagner, M., Steglich, S. and Kellerer, W. (2005) 'Situational Reasoning – a Practical OWL Use Case' *Proceedings of the 7th International Symposium on Autonomous Decentralized Systems (ISADS'05)*, Chengdu, China, April.

Mrohs, B., Luther, M. and Vaidya, R. (2005) 'Context-aware Presence Management' *Proceedings of the Workshop*

on Context Awareness for Proactive Systems (CAPS'05), pp. 100–103, Helsinki, Finland, June.
Mrohs, B., Luther, M., Vaidya, R., Wagner, M., Steglich, S., Kellerer, W. and Arbanowski, S. (2005) 'OWL-SF – a Distributed Semantic Service Framework' *Proceedings of the Workshop on Context Awareness for Proactive Systems (CAPS'05)*, pp. 67–77, Helsinki, Finland, June.

第 8 章 语义服务

参考文献

[1] NTT DoCoMo Inc. (2002) *I-Mode Service Guideline*, Version 1.2.0, 4 March [http://www.nttdocomo.com/technologies/present/imodetechnology/disclaimer.html].
[2] Booth, D., Liu, C. (2007) Web Services Description Language (WSDL) Version 2.0 Part 0: Primer; W3C Recommendation 26 June 2007 [http://www.w3.org/TR/wsdl20-primer]
[3] Mitra, N., Lafon, Y. (2007) SOAP Version 1.2 Part 0: Primer (Second Edition) [http://www.w3.org/TR/soap12-part0/]
[4] OASIS (2000) *The UDDI Technical White Paper* Technical report. OASIS.
[5] Booth, D., Haas, H., McCabe, F., Newcomer, E., Champion, M., Ferris, C. and Orchard, D. (2004) *Web Services Architecture*. W3C Working Group Note [http://www.w3.org/TR/ws-arch].
[6] Berners-Lee, T., Hendler, J. and Lassila, O. (2001) 'The Semantic Web. *Scientific American* **284**(5), 34–43.
[7] Martin, D., Burstein, M., Hobbs, J., Lassila, O., McDermott, D., McIlraith, S., Narayanan, S., Paolucci, M., Parsia, B., Payne, T., Sirin, E., Srinivasan, N. and Sycara, K. (2004) *Owl-s: Semantic Markup for Web Services*. Member Submission, W3C. World Wide Web Consortium.
[8] Open Mobile Alliance (OMA) (2004) *OMA Web Services Enabler (OWSER): Overview*.
[9] ETSI-3GPP (2005) *Universal Mobile Telecommunications System (UMTS); Open Service Access (OSA); Parlay X Web Services; Part 4: Short Messaging (3GPP TS 29.199-04 version 6.3.0 Release 6)*. Technical Specification.
[10] Colgrave J. and Januszewski, K. (undated) *Using WSDL in a UDDI Registry, version 2.0.2* Technical Note. OASIS.
[11] Dean, M., Schreiber, G., Bechhofer, S., van Harmelen, F., Hendler, J., Horrocks, I., McGuinness, D.L., Patel-Schneider, P.F. and Stein, L.A. (2004) *OWL Web Ontology Language Reference, 2004*. W3C Recommendation. World Wide Web Consortium [http://www.w3.org/TR/owl-ref].
[12] Baader, F., Calvanese, D., McGuinness, D., Nardi, D. and Patel-Schneider, P. (2003) *The Description Logic Handbook – Theory, Implementation and Applications*. Cambridge University Press.
[13] Paolucci, M., Ankolekar, A., Srinivasan, M. and Sycara, K. (2003) 'The DAML-S Virtual Machine' *Proceedings of the Second International Semantic Web Conference*, Sanibel Island, FL.
[14] Masuoka, R., Labrou, Y., Parsia, B. and Sirin, E. (2003) 'Ontology-enabled Pervasive Computing Applications' *IEEE Intelligent Systems* **18**(5), 68–72.
[15] Lawrence, S. (2002) *Basic Device Definition version 1.0*. UPnP Standard.
[16] Paolucci, M., Goix, W., Andreetto, A., Luther, M. and Wagner, M. (2005) 'Representing Services for Mobile Computing using OWL and OWL-S: An Initial Investigation' *Proceedings of Web Service Composition Workshop (wscomps05)*.
[17] Denker, G., Kagal, L., Finin, T., Paolucci, M., Srinivasan, N. and Sycara, K. (2003) 'Security for DAML Web services: Annotation and Matchmaking' *Proceedings of the Second International Semantic Web Conference (ISWC 2003)*.
[18] Paolucci, M., Kawamura, T., Payne, T.R. and Sycara, K. (2002) 'Semantic Matching of Web Services Capabilities' *Proceedings of the First International Semantic Web Conference*.
[19] Sycara, K., Paolucci, M., Anolekar, A. and Srinivasan, N. (2003) 'Automated Discovery, Interaction and Composition of Semantic Web Services' *Web Semantics* **1**(1).
[20] Li L. and Horrocks, I. (2003) 'E-commerce: A Software Framework for Matchmaking based on Semantic Web Technology' *Proceedings of the Twelfth International Conference on World Wide Web*, Budapest, Hungary.
[21] Mandell, D. and McIllraith, S. (2003) 'A Bottom-up Approach to Automating Web Service Discovery, Customization, and Semantic Translation' *Proceedings of the 12th International Conference on the World Wide Web (WWW 2003)*. ACM Press.
[22] Colucci, S., Noia, T.D., Sciascio, E.D., Donini, F. and Mongiello, M. (2004) 'Concept Abduction and Contraction for Semantic-based Discovery of Matches and Negotiation Spaces in an e-Marketplace' *Proceedings of*

the 6th International Conference on Electronic Commerce (ICEC 2004). ACM Press.

[23] Constantinescu, I. and Faltings, B. (2003) 'Efficient Matchmaking and Directory Services' *Proceedings of IEEE/WIC International Conference on Web Intelligence.*

[24] Klein, M. and Koenig-Ries, B. (2004) 'Coupled Signature and Specification Matching for Automatic Service Binding' *Procedings of European Conference on Web Services*, LNAI, pp. 183–197. Springer.

[25] Banaei-Kashani, F., Chen, C.-C. and Shahabi, C. (2004) 'Wspds: Web services Peer-to-peer Discovery Service' *Proceedings of International Symposium on Web Services and Applications (ISWS).*

[26] Mallya, A.U., Desai, N., Chopra, A.K. and Singh, M.P. (2005) 'Owl-p: OWL for Protocols and Processes' *Proceedings of the Fourth International Conference on Autonomous Agents and MultiAgent Systems.*

[27] Peer, J. (2005) 'Semantic Service Markup with Sesma' *Proceedings of the Web Service Semantics Workshop (WSS'05) at WWW'05.*

[28] Confalonieri, R., Domingue, J. and Motta, E. (2004) 'Orchestration of Semantic Web services in IRS-III' *Proceedings of the First AKT Workshop on Semantic Web Services (AKT-SWS'04)*, Milton Keynes, UK, December.

[29] Hakimpour, F., Confalonieri, R., Sell, D. and Domingue, J. (2005) 'Orchestration of WSMO-based Semantic Web Services in IRS-III' *Proceedings of the 2nd European Semantic Web Conference (ESWC'05)*, Heraklion, Greece.

[30] Bouquet, P., Giunchiglia, F., van Harmelen, F., Serafini, L. and Stuckenschmidt, H. (2003) 'C-owl: Contextualizing Ontologies' *Second International Semantic Web Conference*, Sanibel Island, FL.

[31] Fensel, D. and Bussler, C. (2002) 'The Web Service Modeling Framework (WSMF)' *Electronic Commerce: Research and Applications* **1**(2), 113–137.

[32] Akkiraju, R., Farrell, J., Miller, J., Nagarajan, M., Schmidt, M.-T., Sheth A., Verma, K. (2005) Web Service Semantics—WSDL-S—Version 1.0 Technical Note, April 2005, World Wide Consortium [http://www.w3.org/Submission/WSDL-S/]

[33] Kopecký, Jacek, Moran, Matthew, Vitvar, Tomas, Roman, Dumitru and Mocan, Adrian (undated) *WSMO Grounding* [http://www.wsmo.org/TR/d24/d24.2/v0.1].

[34] Paolucci, Massimo, Wagner, Matthias and Martin, David (2007) 'Grounding OWL-S in SAWSDL' *Proceedings of the Fifth International Conference on Service-Oriented Computing*, Vienna, Austria, September, pp. 416–421.

[35] Martin, David, Paolucci, Massimo and Wagner, Matthias (2007) 'Bringing Semantic Annotation to Web services: OWL-S from the SAWSDL Perspective' *Proceedings of the 6th International Semantic Web Conference*, Busan, Korea, November.

[36] Chen, H., Perich, F., Finin, T. and Joshi, A. (2004) 'SOUPA: Standard Ontology for Ubiquitous and Pervasive Applications' *International Conference on Mobile and Ubiquitous Systems: Networking and Services*, pp. 258–267, Boston, MA, August.

[37] Chen, H., Perich, F., Chakraborty, D., Finin, T. and Joshi, A. (2004) 'Intelligent Agents meet Semantic Web in a Smart Meeting Room' *Proceedings of the Third International Joint Conference on Autonomous Agents & Multi Agent Systems (AAMAS'04)*, New York City, NY, July.

[38] Mrohs, B., Luther, M., Vaidya, R., Wagner, M., Steglich, S., Kellerer, W. and Arbanowski, S. (2005) 'Owl-sf – A Distributed Semantic Service Framework' *Proceedings of Workshop on Context Awareness for Proactive Systems (CAPS 2005)*, pp. 67–77.

[39] Gandon, F. and Sadeh, N. (2004) 'Semantic Web Technologies to Reconcile Privacy and Context Awareness' *Web Semantics Journal* **1**(3).

[40] Noppens, O., Liebig, Th., Schmidt, P., Luther, M. and Wagner, M. 'MobiXPL – A SVG-based Mobile User Interface for Semantic Service Discovery' *Proceedings of the 5th International Conference on Scalable Vector Graphics (SVGOPEN'07)*, Tokyo, Japan, September.

[41] Belecheanu, R., Jawaheer, G., Hoskins, A., McCann, J.A. and Payne, T. (2004) 'Semantic Web Meets Autonomic Ubicomp' *Proceedings of the 3rd International* Semantic Web Conference.

[42] Vaculin, Roman and Sycara, Katia (2007) 'Specifying and Monitoring Composite Events for Semantic Web Services' *The 5th IEEE European Conference on Web Services*. IEEE Computer Society.

[43] Andrews, T., Curbera, F., Dholakia, H., Goland, Y., Klein, J., Leymann, F., Liu, K., Roller, D., Smith, D., Thatte, S., Trickovic, I. and Weerawarana, S. (2003) *Specification: Business Process Execution Language for Web Services, version 1.1* [http://www.ibm.com/developerworks/library/ws-bpel].

[44] Burdett, Kavantzas, N. (2004) WS Choreography Model Overview. W3C Working Draft. World Wide Web Consortium. [http://www.w3.org/TR/ws-chor-model/]

[45] Sirin, Evren, Parsia, Bijan, Wu, Dan, Hendler, James and Nau, Dana (2004) 'HTN Planning for Web Service

Composition Using SHOP2' *Journal of Web Semantics* **1**(4), 377–396.
[46] Ghallab, M., Nau, D. and Traverso P. (2004) *Automated Planning*. Elsevier.

第 9 章　动态适配——实时调整服务

参考文献

[1] Filman, Robert E., Friedman, Daniel P. (2005) 'Aspect-Oriented Programming Is Quantification and Obliviousness' in Filman, Robert E., Elrad, Tzilla, Clarke, Siobhan, Akşit, Mehmet (eds.), *Aspect-Oriented Software Development*, pp. 21–35, Addison-Wesley.
[2] Filman, Robert E. (2001) 'What is Aspect-Oriented Programming, Revisited' ECOOP 2001 Workshop on Advanced Separation of Concerns [http://trese.cs.utwente.nl/Workshops/ecoop01asoc/papers/Filman.pdf].
[3] Kiczales, Gregor, Lamping, John, Mendhekar, Anurag, Maeda, Chris, Lopes, Cristina, Loingtier, Jean-Marc and Irwin, John (1997) 'Aspect-Oriented Programming' in Akšit, Mehmet and Matsuoka, Satoshi (eds), *Proceedings of the European Conference on Object-Oriented Programming*, LNCS 1241, pp. 220–242. Springer.
[4] Videira Lopes, Cristina (2002) *Aspect-Oriented Programming: An Historical Perspective (What's in a Name?)*. Report UCI-ISR-02-5, University of California, Irvine, CA.
[5] Parnas, David L. (1972) 'On the Criteria to be used in Decomposing Systems into Modules' *Communications of the ACM* **15**(12), 1053–1058.
[6] Pree, Wolfgang (1995) *Design Patterns for Object-Oriented Software Development*. Addison-Wesley.
[7] Ernst, Erik (2003) 'Separation of Concerns' *AOSD 2003 Workshop on Software-Engineering Properties of Languages for Aspect Technologies (SPLAT)}*, Boston, MA, March.
[8] Gosling, James, Joy, Bill, Steele, Guy and Bracha, Gilad (2000) *The Java Language Specification* (2nd edn). Addison-Wesley.
[9] Kniesel, Günter, Costanza, Pascal and Austermann, Michael (2001) 'JMangler – A Framework for Load-Time Transformation of {Java} Class Files' *First IEEE International Workshop on Source Code Analysis and Manipulation (SCAM 2001)*, Florence, Italy, November. IEEE Computer Society Press [http://www.informatik.uni-bonn.de/\~costanza/SCAM_jmangler.pdf]
[10] Hirschfeld, Robert (2003) 'AspectS – Aspect-Oriented Programming with Squeak' In Akšit, Mehmet, Mezini, Mira and Unland, Rainer (eds), *Objects, Components, Architectures, Services, and Applications for a Networked World*, LNCS 2591, pp. 216–232. Springer.
[11] Brant, John, Foote, Brian, Johnson, Ralph E. and Roberts, Don (1998) 'Wrappers to the Rescue' *Proceedings of the European Conference on Object-Oriented Programming*, LNCS 1445, pp. 396–417. Springer.
[12] Maes, Pattie (1987) *Computational Reflection*. Artificial Intelligence Laboratory, University of Brussels (VUB).
[13] Kiczales, Gregor, des Rivieres, Jim and Bobrow, Daniel G. (1991) *The Art of the Metaobject Protocol*. Addison-Wesley.
[14] Mendhekar, Anurag, Kiczales, Gregor and Lamping, John (1997) *RG: A Case-Study for Aspect-Oriented Programming* Report SPL97-009 P9710044. Xerox PARC.
[15] Lopes, Cristina Videira (1997) *D: A Language Framework for Distributed Programming* Dissertation, College of Computer Science, Northeastern University [http://www.parc.xerox.com/csl/groups/sda/pubs/papers/Lopes-Thesis/dissertation.pdf].
[16] Kay, Alan (2002) *Is Software Engineering an Oxymoron?* Viewpoints Research Institute, Glendale, CA.
[17] Czarnecki, Krzysztof (1998) *Generative Programming: Principles and Techniques of Software Engineering Based on Automated Configuration and Fragment-Based Component Models* PhD Thesis, Technical University of Ilmenau.
[18] Goldberg, Adele and Robson, David (1983) *Smalltalk-80: The Language and its Implementation*. Addison-Wesley.
[19] Hirschfeld, Robert and Wagner, Matthias (2002) 'PerspectiveS – AspectS with Context' *OOPSLA 2002 Workshop on Engineering Context-Aware Object-Oriented Systems and Environments (ECOOSE)*, Seattle, WA, November.
[20] Hirschfeld, Robert, Costanza, Pascal and Yierstrast, Oscar (2008) 'Context-oriented Programming' *Journal of Object Technology* April 2008, www.jot.fm.

延伸阅读

Aßmann, Uwe (2003) *Invasive Software Composition*. Springer.

Bracha, Gilad and Cook, William (1990) 'Mixin-based *Inheritance*' *Proceedings of the European Conference on Object-Oriented Programming on Object-Oriented Programming Systems, Languages, and Applications*, Ottawa, ON, Canada, pp. 303–311. ACM Press [http://doi.acm.org/10.1145/97945.97982]

Canning, Peter, Cook, William, Hill, Walter, Olthoff, Walter and Mitchell, John C. (1989) 'F-bounded Polymorphism for Object-Oriented Programming' *Proceedings of the International Conference on Functional Programming Languages and Computer Architecture*, London, September, pp. 273–280. ACM Press [http://doi.acm.org/10.1145/99370.99392]

Clarke, Siobhàn and Walker, Robert (2002) 'Towards a Standard Design Language for AOSD' In Kiczales, Gregor (ed.), *Proceedings of the International Conference on Aspect-Oriented Software Development (AOSD 2002)*, Enschede, The Netherlands, April, pp. 113–119. ACM Press.

Dijkstra, Edsger W. (1976) *A Discipline of Programming*. Prentice-Hall.

Ernst Erik and Lorenz, David H. (2003) 'Aspects and Polymorphism in AspectJ' In Akšit, Mehmet *Proceedings of the International Conference on Aspect-Oriented Software Development (AOSD 2003)*, Boston, MA, March, pp. 150–157. ACM Press.

Gamma, Erich, Helm, Richard, Johnson, Ralph and Vlissides, John (1995) *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley.

Hanenberg, Stefan, Hirschfeld, Robert and Unland, Rainer (2003) 'Aspect Weaving: Using the Base Language's Introspective Facilities to Determine Join Points' *Workshop on Advancing the State-of-the-Art in Runtime Inspection (ECOOP 2003)*, Darmstadt, Germany, July [http://www.st.informatik.tu-darmstadt.de/pages/workshops/ASARTI03/HanenbergASARTI03.pdf].

Hanenberg, Stefan and Unland, Rainer (2001) 'Using and Reusing Aspects in AspectJ' *Workshop on Advanced Separation of Concerns in Object-Oriented Systems (OOPSLA 2001)*, Tampa, FL, October, pp 80–89 [http://www.cs.ubc.ca/\~kdvolder/Workshops/OOPSLA2001/submissions/11-hanenberg.pdf]

Hanenberg, Stefan and Unland, Rainer (2002) 'Roles and Aspects: Similarities, Differences, and Synergetic Potential' In Bellahsene, Zohra, Patel, Dilip and Rolland, Colette (eds), *Object-Oriented Information Systems* LNCS 2425, pp. 507–521. Springer.

Ingalls, Dan, Kaehler, Ted, Maloney, John, Wallace, Scott and Kay, Alan (1997) 'Back to the Future: The Story of Squeak, a Practical Smalltalk Written in Itself' *Proceedings of the Conference on Object-Oriented Programming, Systems, Languages, and Applications*, Atlanta, GA, October, pp. 318–326. ACM Press [http://doi.acm.org/10.1145/263698.263754].

Holland, Ian M. (1992) 'Specifying Reusable Components using C*ontracts' Proceedings of the European Conference on Object-Oriented Programming*, Utrecht, The Netherlands, LNCS 615, pp. 287–308. Springer.

Jézéquel, Jean-Marc, Plouzeau, Noël, Weis, Torben and Geihs, Kurt (2002) 'From Contracts to Aspects in UML Designs' In Aldawud, Omar, Booch, Grady, Clarke, Siobhàn, Elrad, Tzilla, Harrison, Bill, Kandi, Mohamed and Strohmeier, Alfred (eds), *Workshop on Aspect-Oriented Modeling with UML (AOSD 2002)*, Enschede, The Netherlands, March [http://lglwww.epfl.ch/workshops/aosd-uml/Allsubs/jean.pdf].

Johnson, Ralph and Foote, Brian (1988) 'Designing Reusable Classes' *Journal of Object-Oriented Programming* **1**(2), 25–35.

Keene, Sonya E. (1989) *Object-Oriented Programming in Common Lisp: A Programmer's Guide to CLOS*. Addison-Wesley.

Kiczales, Gregor, Hilsdale, Erik, Hugunin, Jim, Kersten, Mik, Palm, Jeffrey and Griswold, William G. (2001) 'An overview of AspectJ' In Lindskov Knudsen, J. (ed.), *Proceedings of the European Conference on Object-Oriented Programming*, LNCS 2072, pp. 327–353. Springer.

Kristensen, Bent B. and Østerbye, Kasper (1996) 'Roles: Conceptual Abstraction Theory and Practical Language Issues' *Theory and Practice of Object Systems* **2**(3), 143–160.

Lopes, Cristina Videira, Dourish, Paul, Lorenz, David H. and Lieberherr, Karl (2003) 'Beyond AOP: Toward Naturalistic Programming' *Companion of the Conference on Object-Oriented Programming, Systems, Languages, and Applications*, Anaheim, CA, pp. 198–207. ACM Press [http://doi.acm.org/10.1145/949344.949400].

Mezini, Mira and Ostermann, Klaus (2002) 'Integrating Independent Components with On-demand Remodularization' *Proceedings of Conference on Object-Oriented Programming, Systems, Languages, and Applications*, Seattle, WA, pp. 52–67. ACM Press [http://doi.acm.org/10.1145/582419.582426].

Orleans, Doug and Lieberherr, Karl (2001) 'DJ: Dynamic Adaptive Programming in Java' In Yonezawa, Akinori

and Matsuoka, Satoshi (eds), *International Conference on Metalevel Architectures and Separation of Crosscutting Concerns (Reflection 2001)*, Kyoto, Japan, September, LNCS 2192, pp. 73–80. Springer.

Pulvermüller, Elke, Speck, Andreas and Rashid, Awais (2000) 'Implementing Collaboration-based Designs using Aspect-Oriented Programming' *Proceedings of TOOLS-USA*, Santa Barbara, CA, July, pp. 95–104.

Smaragdakis, Yannis and Batory, Don (1998) 'Implementing Reusable Object-Oriented Components' *Proceedings of the International Conference on Software Reuse*, Victoria, BC, Canada, June, pp. 36–45. IEEE Computer Society Press [http://citeseer.nj.nec.com/article/smaragdakis98implementing.html]

VanHilst, Michael and Notkin, David (1996) 'Using Role Components to Implement Collaboration-Based Designs' *Proceedings of Conference on Object-Oriented Programming, Systems, Languages, and Applications*, San Jose, CA, October, pp. 359–369. ACM Press [http://doi.acm.org/10.1145/236337.236375, [http://citeseer.nj.nec.com/vanhilst96using.html].

Vlissides, John M. (1996) 'Protection, Part I: The Hollywood Principle' *C++ Report*, February [http://www.squeak.org].

第 10 章　上下文感知的移动性管理

参考文献

[1] Wei, Q., Farkas, K., Mendes, P., Prehofer, C., Plattner, B. and Nafisi, N. (2003) 'Context-aware Handover Based on Active Network Technology' *IWAN 2003*. Springer.

[2] Prehofer, Christian, Kellerer, Wolfgang, Hirschfeld, Robert, Berndt, Hendrik and Kawamura, Katsuya (2002) 'An Architecture Supporting Adaptation and Evolution in Fourth Generation Mobile Communication Systems' *Journal of Communications and Networks* **4**(4).

[3] Prehofer, C. and Wei, Q. (2002) 'Active Networks for 4G Mobile Communication: Motivation, Architecture and Application Scenarios' *IWAN 2002*. Springer.

[4] Kempf, J. (2001) 'Dormant Mode Host Alerting ("IP paging") problem statement' RFC3132, June.

[5] Castelluccia, C. (2001) 'Extending Mobile IP with Adaptive Individual Paging: A Performance Analysis' *ACM Mobile Computing and Communications Review (MC2R)*, April.

[6] Hsiao-Kuang Wu, Ming-Hui Jin, Jorng-Tzong Horng and Cheng-Yi Ke (2001) 'Personal Paging Area Based On Mobile's Moving Behaviours' *IEEE INFOCOM*, May.

[7] Lei, Z., Sarazdar, C.U. and Mandayam, N.B. (1999) 'Mobility Parameter Estimation for the Optimization of Personal Paging Areas in PCS/Cellular Mobile Networks' *Proceedings of the 2nd IEEE Signal Processing Workshop on Signal Processing Advances in Wireless Communications (SPAWC'99)*, 9–12 May.

[8] Stemm, M. and Katz, R. (1998) 'Vertical Handoffs in Wireless Overlay Networks' *ACM Journal on Mobile Networks and Applications* **3**(4).

[9] Pahlavan, K., Krishnamurthy, P., Hatami, A., Ylianttila, M., Makela, J.P., Pichna, R., Vallstron J. (2000) 'Handoff in Hybrid Mobile Data Networks' *IEEE Communication Magazine*, April.

[10] Chan, P.M.L., Sheriff, R.E., Conforto, P., Tocci, C. (2001) 'Mobility Management Incorporating Fuzzy Logic for a Heterogeneous IP Environment' *IEEE Communication Magazine*, December.

[11] Kounavis, Michael E., Campbell, Andrew T., Ito, G. and Bianchi, G. (2001) 'Design, Implementation and Evaluation of Programmable Handoff in Mobile Networks' *Mobile Networks and Applications* **6**, 443–461.

[12] Prehofer, C., Nafisi, N. and Wei, Q. (2003) 'A Framework for Context-aware Handover Decisions' *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, Beijing, China, September.

[13] *Composite Capability/Preference Profiles (CC/PP)* (2004) W3C Working Draft. World Wide Web Consortium [http://www.w3.org/Mobile/CCPP].

[14] Psounis, K. (1999) 'Active Networks: Applications, Security, Safety, and Architectures' *IEEE Communications Surveys*, First Quarter.

[15] Mendes, Paulo, Prehofer, Christian and Wei, Qing (2003) 'Context Management with Programmable Mobile Networks' *IEEE Computer Communication Workshop.*

[16] Prehofer, C. and Wei, Q. (2002) 'Active Networks for 4G Mobile Communication: Motivation, Architecture and Application Scenarios' *International Working Conference on Active Networks*, Zurich, Switzerland.

[17] Keller, R., Ruf, L., Guindehi, A. and Plattner, B. (2002) 'PromethOS: A Dynamically Extensible Router Architecture Supporting Explicit Routing' *IWAN 2002*, December, Springer.

[18] Bossardt, Matthias, Hoog Antink, Roman, Moser, Andreas and Plattner, Bernhard (2003) 'Chameleon:

Realizing Automatic Service Composition for Extensible Active Routers' *Proceedings of Fifth Annual International Working Conference on Active Networks (IWAN 2003)*, Kyoto, Japan, December, LNCS. Springer.

[19] Wang, Helen J., Katz, Randy H. and Giese, Jochen (1999) 'Policy-enabled Handoffs across Heterogeneous Wireless Networks' *WMCSA 99*, February. IEEE.

[20] Dey, A. (2000) *Providing Architectural Support for Building Context-Aware Applications* PhD thesis, College of Computing, Georgia Institute of Technology.

[21] Chen, G. and Kotz, D. (2002) 'Context Aggregation and Distribution in Ubiquitous Computing Systems' *Proceedings of IEEE Workshop on Mobile Computing Systems and Applications*, Callicoon, NY, February.

延伸阅读

Anetd: Active Networks Daemon (undated) ACTIVE project. ISI & SRI. [http://www.sdl.sri.com/projects/activate/anted].

MGEN – Multi-Generator Toolset (undated) [http://mgen.pf.itd.nrl.navy.mil].

第11章 智能上下文

参考文献

[1] Mrohs, B., Luther, M., Vaidya, R., Wagner, M., Steglich, S., Kellerer, W. and Arbanowski, S. (2005) 'OWL-SF – a Distributed Semantic Service Framework' *Proceedings of the Workshop on Context Awareness for Proactive Systems (CAPS'05)*, Helsinki, Finland, June, pp. 67–77.

[2] McGuinness D. and van Harmelen, F. (2004) *OWL Web Ontology Language Overview* W3C Recommendation. Wide Web Consortium.

[3] Sameshima, S., Suzuki, J. (2004) *Platform Independent Model and Platform Specific Model for SDOs* Final recommended specification. OMG.

[4] Fielding. T.R. (2000) *Architectural Styles and the Design of Network-based Software Architectures*. PhD thesis, University of California, Irvine.

[5] Mrohs, B., Luther, M. and Vaidya, R. (2005) 'Context-aware Presence Management' *Proceedings of the Workshop on Context Awareness for Proactive Systems (CAPS'05)*, June.

[6] Dey, A.K. (2000) *Providing Architectural Support for Building Context-aware Applications*. PhD thesis, Georgia Institute of Technology.

[7] Chen, H., Finin, T. and Joshi, A. (2004) 'A Context Broker for Building Smart Meeting Rooms. *Proceedings of the Autonomous Systems Symposium*, AAAI Spring Symposium. CA, March.

[8] Khushraj, D. and Lassila, O. (2004) 'CALI: Context Awareness via Logical Inference' *Proceedings of the Workshop on Semantic Web Technology for Mobile and Ubiquitous Applications*, November.

[9] ESSI WSML Working Group. Web Services modeling Language WSML. http://www.wsmo.org/wsml

[10] MacGregor, R. (1991) 'Using a Description Classifier to Enhance Deductive Inference' *Proceedings of the 7th IEEE Conference on AI Applications*, pp. 141–147.

[11] Grosof, B.N., Horrocks, I., Volz, R. and Decker, Stefan (2003) 'Combining Logic Programs with Description Logic' *Proceedings of the 12th International World Wide Web Conference*. ACM.

[12] Fensel, D. and Bussler, C. (2005) 'The Web Service Modeling Framework WSMF' *International Journal of Electronic Commerce* **9**(2).

[13] Paolucci, M., Goix, W., Andreetto, A., Luther M. and Wagner M. (2005) 'Representing Services for Mobile Computing using OWL and OWL-S: An Initial Investigation' *Proceedings of the Workshop on Web service Composition in conjunction with the International Conference on Web Intelligence and Intelligent Agent Technology*, Compiège, France, September 2005.

[14] Beydoun, G. and Hoffmann, A. (2000) 'Monitoring Knowledge Acquisition, Instead of Evaluating Knowledge Bases' *Proceedings of the European Knowledge Acquisition Conference (EKAW2000)*, Juan-les-Pins, France, LNCS 1937. Springer.

[15] W3C. Web Ontology Language (OWL) http://www.w3.org/2004/OWL

延伸阅读

3GPP (2002) *The Third Generation Partnership Project* [http://www.3gpp.org].

Balke, W.-T. and Wagner. M. (2003) 'Cooperative Discovery for User-centered Web Service Provisioning' Proceedings of the 1st International Conference on Web Services (ICWS'03), Las Vegas.

Balke, W.-T. and Wagner, M. (2003) 'Towards Personalized Selection of Web Services' *Proceedings of the Interna-*

tional World Wide Web Conference (WWW), Budapest, Hungary.

Balke, W.-T. and Wagner, M. (2004) 'Through Different Eyes – Assessing Multiple Conceptual Views for Querying Web Services' Proceedings of the 13th International. World Wide Web Conerence. (WWW2004) Alternate Track on Web Services, New York.

Banerji, A., Bartolini, C., Beringer, D., Chopella, V., Govindarajan, K., Karp, A., Kuno, H., Lemon, M., Pogossiants, G., Sharma, S. and Williams, S. (2002) *Web Services Conversation Language (WSCL) 1.0* [http://www.w3.org/TR/wscl10].

Berners-Lee, T., Hendler, J. and Lassila. O. (2001) 'The Semantic Web' *Scientific American*, **284**(5), 34–43.

Casati, F. and Shan. M. (2001) 'Dynamic and Adaptive Composition of E-Services' *Journal of Information Systems* **6**, 143–163.

Chu, H., Yang, K., Chiang, M. Minock, Chow, G. and Larson, C. (1996) 'CoBase: A Scalable and Extensible Cooperative Information System' *Journal of Intelligent Information Systems (JIIS)* **6**(3), 223–259.

IETF and World Wide Web Consortium (2002) *XML Signature* [http://www.w3.org/Signature].

Kießling, W. and Köstler, G. (2002) 'Preference SQL – Design, Implementation, Experiences' *Proceedings of the International Conference on Very Large Data Bases (VLDB'02)*, Hong Kong, China.

Leymann, F. (2001) *Web Services Flow Language (WSFL 1.0)* [http://www-4.ibm.com/software/solutions/webservices/pdf/WSFL.pdf].

Luther, M., Mrohs, B., Wagner, M., Steglich, S. and Kellerer, W. (2005) 'Situational Reasoning – A Practical OWL Use Case' *Proceedings of the 7th International Symposium on Autonomous Decentralized Systems (ISADS'05)*, Chengdu, China, April.

Minker, J. (1998) 'An Overview of Cooperative Answering in Data-bases' *Proceedings of the Inernational. Conference on Flexible Query Answering Systems (FQAS)*, Roskilde, Denmark, LNCS 1495. Springer.

Motro, A. (1988) 'VAGUE: A User Interface to Relational Databases that Permits Vague Queries' *ACM Transactions on Office Information Systems (TOIS)*, **6**, 187–214.

Narayanan, S. and McIlraith, S. (2002) 'Simulation, Verification and Automated Composition of Web Services' *Proceedings of the International World Wide Web Conference (WWW 2002)*, Honolulu, Hawaii, pp. 77–88.

Paolucci, M., Kawamura, T., Payne, T. and Sycara, K. (2002) 'Semantic Matching of Web Services Capabilities' *Proceedings of the International Semantic Web Conference (ISWC'02)*, Sardinia, Italy.

Paolucci, M., Kawamura, T., Payne, T. and Sycara, K. (2002) 'Importing the Semantic Web in UDDI' *Proceedings of the International Workshop on Web Services, e-Business and the Semantic Web (WES'02)*, Toronto, Canada.

Parlay Group, The (undated) *Parlay/OSA APIs* [http://www.parlay.org].

Pires, P., Benevides, M. and Mattoso, M. (2002) 'Building Reliable Web Services Compositions' *Proceedings of the International Workshop on Web Services: Research, Standardization and Deployment (WS-RSD)*, Erfurt, Germany, pp. 551–562.

Thatte, S. (2001) *XLANG: Web Services for Business Process Design* [http://www.gotdotnet.com/team/xml_wsspecs/xlang-c/default.htm].

Vilain, M. (1990) 'Getting Serious about Parsing Plans: A Grammatical Analysis of Plan Recognition' *Proceedings of the National Conference on Artificial Intelligence (AAAI-90)*, Boston, pp. 190–197.

Wagner, M., Balke, W.-T., Hirschfeld, R. and Kellerer, W. (2002) 'A Roadmap to Advanced Personalization of Mobile Services' *Proceedings of the International Conference DOA/ODBASE/ CoopIS (Industry Program)*, Irvine, CA.

Wagner, M., Kießling, W. and Balke, W.-T. (2002) 'Progressive Content Delivery for Mobile E-Services' *Proceedings of the 3rd International Conferenc. on Advances in Web-Age Information Management (WAIM2002)*, Beijing, China, LNCS 2419, pp. 225–235. Springer.

第 12 章　从个人移动性到移动个性化

参考文献

[1] Wagner, M., Luther, M., Hirschfeld, R. , Kellerer, W. and Tarlano, A. (2005) 'From Personal Mobility to Mobile Personality' *Telenor Telektronikk Magazine*, Special Issue on Future Mobile Phone.

[2] Wagner, M., Balke, W.-T., Hirschfeld, R. and Kellerer, W. (2002) 'A Roadmap to Advanced Personalization of Mobile *Services*' *Proceedings of the International Conference DOA/ODBASE/ CoopIS (Industry Program)*, Irvine, CA, October.

[3] Kellerer, W., Wagner, W. and Balke, W.-T. (2003) 'Preference-based Session Management' *Proceedings of the 8th International Workshop on Mobile Multimedia Communications (MOMUC2003)*, Munich, Germany, October.

[4] Lassila, O. and Swick, R.R. (1999) *Resource Description Format: Model and Syntax Specification*. W3C Recommendation. World Wide Web Consortium.

[5] Berners-Lee, T., Hendler, J. and Lassila, O. (2001) 'The Semantic Web' *Scientific American*, **284**(5), 34–43.

[6] McGuinness, D.L. and van Harmelen, F. (2003) *OWL Web Ontology Language Overview*. W3C Working Draft. World Wide Web Consortium [http://www.w3.org/TR/owl-features].

[7] Balke, W.-T. and Wagner, M. (2003) 'Towards Personalized Selection of Web Services' *Proceedings of the International World Wide Web Conference (WWW2003)*, Budapest, Hungary, May.

[8] Balke, W.-T. and Wagner, M. (2003) 'Cooperative Discovery for User-centered Web Service Provisioning' *Proceedings of the 1st International Web Service Conference (ICWS2003)*, Las Vegas, NV.

[9] Naghshineh, M. (2002) 'Context-Aware Computing' *IEEE Wireless Communications Magazine*, Special Issue, October.

[10] Floreen, P., Przybilski, M., Nurmi, P., Koolwaaij, J., Tarlano, A., Wagner, M., Luther, M., Bataille, F., Boussard, M., Mrohs, B. and Lau, S. (2005) 'Towards a Context Management Framework for MobiLife' *Proceedings of the 14th IST Mobile & Wireless Communications Summit*, Dresden, Germany, 19–23 June.

[11] Koolwaaij, Johan, Tarlano, Anthony, Luther, Marko, Nurmi, Petteri, Mrohs, Bernd, Battestini, Agathe and Vaidya, Raju (2006) 'Context Watcher – Sharing Context Information in Everyday Life' *Proceedings of the IASTED International Conference on Web Technologies, Applications, and Services (WTAS2006)*, Calgary, Canada, July.

[12] Kellerer, W. and Berndt, H. (2002) 'Next Generation Service Session Signaling' *Proceedings of TINA 2002*, Petaling Jaya, Malaysia.

[13] Prehofer, C., Kellerer, W., Hirschfeld, R., Berndt, H. and Kawamura, K. (2002) 'An Architecture Supporting Adaptation and Evolution in Fourth Generation Mobile Communication Systems' *Journal of Communications and Networks* **4**(4).

[14] Hirschfeld, R. and Wagner, M. (2002) 'PerspectiveS – AspectS with Context' *Proceedings of the OOPSLA 2002 Workshop on Engineering Context-Aware Object-Oriented Systems and Environments*, Seattle, WA, November.

[15] Hirschfeld, R. (2002) 'AspectS – Aspect-Oriented Programming with Squeak' *Architectures, Services, and Applications for a Networked World*, LNCS 2591. Springer.

[16] Hirschfeld, R., Wagner, M., Kellerer, W. and Prehofer, C. (2003) 'AOSD for System Integration and Personalization' *Proceedings of the AOSD Workshop on the Commercialization of AOSD Technology*, Boston, MA, March.

[17] Noppens, O., Luther, M., Liebig, T., Wagner, M. and Paolucci, M. (2006) 'Ontology-based Preference Handling for Mobile Music Selection' *Proceedings of the 3rd Workshop on Advances in Preference Handling in conjunction with ECAI'06*, Riva del Garda, Italy.

[18] Wagner, M., Liebig, T., Noppens, O., Balzer, S. and Kellerer, W. (2004) 'mobiXPL – Semantic-based Service Discovery on Tiny Mobile Devices' *Proceedings of Workshop on Semantic Web Technology for Mobile and Ubiquitous Applications (in conjunction with ISWC'04)*, Hiroshima, Japan.

[19] Kießling, W. and Köstler, G. (2002) 'Preference SQL – Design, Implementation, Experiences' *Proceedings of the International Confernce on Very Large Databases (VLDB'02)*, Hong Kong.

[20] Balke, W.-T. and Wagner, M. (2004) 'Through Different Eyes – Assessing Multiple Conceptual Views for Querying Web Services' *Proceedings of 13th International World Wide Web Conference (WWW'04)*, New York.

关于作者

Hendrik Berndt是DoCoMo欧洲实验室的首席技术官（CTO）、高级副主任，他同时也是智能和安全服务以及普适网络研究组织的主管。他主持了面向4G移动系统的研发工作。他曾经担任过电信信息网络架构联盟的首席技术官，美国新泽西Tinton Falls的总负责人，他同时也是TINA架构部门的主席并负责指导技术的研发和多媒体服分布式处理环境以及IP控制和管理等产品的开发工作。他当时在德国电信工业领域担任了多个科研和管理职务，他是美国Reston的Global One公司的高级技术执行总裁，并且被邀请成为Sprints网络和架构计划办公室的成员。

目前他被选为欧洲电子移动基础平台的指导组成员，对象管理组织的指导组成员，Needham的主任，德国柏林开放通信系统管理组成员。他获得了电子工程专业的硕士和博士学位，从2000年开始一直作为访问学者在日本东京的全球信息和电信中心工作。

国际视野 科技前沿

国际信息工程先进技术译丛

《无线Mesh网络架构与协议》

《UMTS蜂窝系统的QoS与QoE管理》

《半导体制造与过程控制基础》

《WCDWA原理与开发设计》

《下一代移动系统：3G/B3G》

《IMS:IP多媒体概念和服务》（原书第2版）

《下一代无线系统与网络》

《深入浅出UMTS无线网络建模、规划与自动优化：理论与实践》

《HSDPA/HSUPA技术与系统设计——第三代移动通信系统宽带无线接入》

《无线传感器及元器件：网络、设计与应用》

《印制电路板——设计、制造、装配与测试》

《IPTV与网络视频：拓展广播电视的应用范围》

《多电压CMOS电路设计》

《微电子技术原理、设计与应用》

《蜂窝网络高级规划与优化2G/2.5G/3G/…向4G的演进》

《基于蜂窝系统的IMS——融合电信领域的VoIP演进》

《无线网络中的合作原理与应用》

《电生理学方法与仪器入门》

《移动电视：DVB-H、DMB、3G系统和富媒体应用》

《环境网络：支持下一代无线业务的多域协同网络》

《基于射频工程的UMTS空中接口设计与网络运行》

《未来UMTS的体系结构与业务平台：全IP的3G CDMA网络》

《UMTS-HSDPA系统的TCP性能》

《宽带无线通信中的空时编码》

《数字图像处理》（原书第4版）

《基于4G系统的移动服务技术》



上架指导：工业技术/通信工程



WILEY
www.wiley.com



ISBN 978-7-111-29117-6

封面设计：马精明

定价：78.00元

地址：北京市百万庄大街22号
邮政编码：100037
电话服务
社服务中心：(010)88361066
销 售 一 部：(010)68326294
销 售 二 部：(010)88379649
读者服务部：(010)68993821
网络服务
门户网：http://www.cmpbook.com
教材网：http://www.cmpedu.com
封面无防伪标均为盗版